



Panduan Pengguna

Amazon Elastic Compute Cloud



Amazon Elastic Compute Cloud: Panduan Pengguna

Copyright © 2024 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 para pelanggan, atau dengan cara apa pun yang menghina 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	11
Langkah 3: Bersihkan instans Anda	14
Langkah selanjutnya	15
Praktik terbaik	17
Amazon Machine Image	20
Menggunakan AMI	21
Membuat AMI Anda sendiri	21
Membeli, berbagi, dan menjual AMI	22
Membatalkan pendaftaran AMI Anda	22
Amazon Linux 2023 dan Amazon Linux 2	23
AMI Windows	23
Jenis AMI	24
Izin peluncuran	24
Penyimpanan untuk perangkat root	25
Tipe virtualisasi	29
Mode boot	32
Meluncurkan instans	34
Parameter mode boot AMI	41
Mode boot tipe instans	43
Mode boot instans	45
Mode boot sistem operasi	47
Mengatur mode boot AMI	49
Variabel UEFI	54
UEFI Secure Boot	55
Temukan AMI	70

Temukan AMI menggunakan konsol Amazon EC2	71
Temukan AMI menggunakan AWS CLI	72
Temukan AMI menggunakan AWS Tools for Windows PowerShell	73
Temukan AMI menggunakan parameter Systems Manager	74
Temukan AMI terbaru menggunakan Systems Manager	78
Informasi lebih lanjut untuk menemukan AMI	80
AMI bersama	80
Penyedia AMI terverifikasi	80
Mencari AMI bersama	81
Menjadikan AMI publik	86
Membagikan AMI dengan organisasi atau unit organisasi	95
Membagikan AMI kepada akun AWS tertentu	105
Membatalkan berbagi AMI dengan akun Anda	110
Menggunakan bookmark	112
Panduan untuk AMI Linux bersama	112
AMI berbayar	119
Menjual AMI Anda	120
Mencari AMI berbayar	120
Membeli AMI berbayar	122
Mendapatkan kode produk untuk instans Anda	123
Menggunakan dukungan berbayar	124
Tagihan untuk AMI berbayar dan didukung	124
Kelola AWS Marketplace langganan Anda	125
Siklus hidup AMI	126
Membuat AMI	126
Memodifikasi AMI	198
Menyalin AMI	198
Menyimpan dan memulihkan AMI	210
Membuat usang sebuah AMI	220
Menonaktifkan AMI	228
Mengarsipkan snapshot AMI	234
Deregister (hapus) AMI	234
Otomatisasi siklus hidup AMI yang didukung EBS	244
Enkripsi AMI	244
Skenario peluncuran instans	245
Skenario penyalinan gambar	248

Memantau peristiwa AMI	251
Peristiwa AMI	252
Buat EventBridge aturan Amazon	255
Memahami penagihan AMI	258
Bidang penagihan AMI	259
Mencari informasi penagihan AMI	261
Memverifikasi biaya AMI pada tagihan Anda	263
Kuota AMI	264
Meminta peningkatan kuota untuk AMI	265
Instans	267
Instans dan AMI	267
Instans	268
AMI	271
Tipe instans	271
Tipe instans yang tersedia	272
Spesifikasi perangkat keras	274
Tipe virtualisasi AMI	276
Menemukan tipe instans	276
Mendapatkan rekomendasi	279
Ubah tipe instans	286
Instans performa yang dapat melonjak	297
Instans GPU	351
Instans Mac	362
Pertimbangan	363
Kesiapan instans	364
AMI macOS EC2	365
EC2 macOS Init	365
Monitor Sistem Amazon EC2 untuk macOS	366
Sumber daya terkait	366
Luncurkan instans Mac	366
Terhubung ke instans Mac Anda	369
Perbarui sistem operasi dan perangkat lunak pada instance Mac	372
Meningkatkan ukuran volume EBS di instans Mac Anda	380
Menghentikan dan mengakhiri instans Mac Anda	381
Temukan versi macOS yang didukung untuk Host Khusus	382
Berlangganan notifikasi AMI macOS	383

Catatan rilis EC2 macOS AMI	385
Optimisasi EBS	387
Tipe instans yang didukung	388
Dapatkan performa maksimum	459
Lihat tipe instans yang mendukung optimisasi EBS	460
Aktifkan optimisasi EBS saat peluncuran	461
Aktifkan optimisasi EBS untuk instans yang sudah ada	462
Opsi pembelian instans	463
Menentukan siklus hidup instans	465
Instans Sesuai Permintaan	466
Instans Terpesan	469
Instans Spot	540
Host Khusus	643
Instans Khusus	705
Reservasi Kapasitas	714
Siklus hidup instans	800
Peluncuran instance	803
Instans berhenti dan mulai	803
Contoh hibernasi	804
Mulai ulang instans	804
Pengakhiran instans	805
Perbedaan antara boot ulang, penghentian, hibernasi, dan pengakhiran	806
Luncurkan	808
Berhenti dan mulai	891
Hibernasi	899
Mulai ulang	931
Mengakhiri	932
Pensiun	943
Ketahanan contoh	948
Cara menggunakan metadata instans	958
Gunakan IMDSv2	959
Mengonfigurasi opsi metadata instans	970
Mengambil metadata instans	994
Bekerja dengan data pengguna instans	1017
Mengambil data dinamis	1021
Kategori metadata instans	1023

Contoh Linux: Nilai indeks peluncuran AMI	1039
Dokumen identitas instans	1044
Peran identitas instans	1109
Menjalankan perintah saat peluncuran	1111
Cara Amazon EC2 menangani data pengguna untuk instans Linux	1111
Bagaimana Amazon EC2 menangani data pengguna untuk instans Windows	1122
Connect ke instans EC2	1136
Hubungkan ke instans Linux Anda	1136
Hubungkan ke instans Windows Anda	1211
Terhubung menggunakan Session Manager	1224
Connect menggunakan EC2 Instance Connect Endpoint	1225
Hubungkan instans Anda ke sumber daya	1252
Identifikasi instans	1296
Periksa sistem UUID	1296
Periksa pengenalan pembuatan mesin virtual sistem	1298
Kelola pengaturan sistem	1303
Mengatur waktu	1304
Kontrol status prosesor	1326
Mengoptimalkan opsi CPU	1328
AMD SEV-SNP	1453
Tambahkan komponen sistem Windows	1459
Kelola pengguna sistem Linux	1464
Atur kata sandi Administrator Windows	1469
Kelola driver perangkat	1470
Menginstal driver NVIDIA	1471
Menginstal driver AMD	1507
Driver Windows PV	1516
AWS Driver Windows NVMe	1552
Konfigurasi instance Windows	1560
Konfigurasi agen peluncuran Windows	1560
Gunakan EC2 Fast Launch untuk Windows	1723
Gunakan akselerator Elastic Graphics di Windows	1748
Instal WSL di Windows	1769
Mutakhirkan instans Windows	1771
Lakukan pemutakhiran langsung	1772
Lakukan pemutakhiran otomatis	1776

Migrasi ke jenis instans generasi saat ini	1788
Migrasikan Microsoft SQL Server dari Windows ke Linux	1798
Memecahkan masalah pemutakhiran	1798
Armada	1799
Armada EC2	1800
Batasan Armada EC2	1802
Instans performa yang dapat melonjak	1802
Tipe permintaan Armada EC2	1803
Strategi konfigurasi Armada EC2	1831
Bekerja dengan Armada EC2	1871
Armada Spot	1899
Tipe permintaan Armada Spot	1899
Strategi konfigurasi Armada Spot	1900
Bekerja dengan Armada Spot	1939
CloudWatch metrik untuk Spot Fleet	1973
Penskalaan otomatis untuk Armada Spot	1977
Memantau peristiwa armada	1987
Tipe peristiwa Armada EC2	1988
Tipe peristiwa Armada Spot	1994
Buat EventBridge aturan	2001
Tutorial	2012
Tutorial: Menggunakan Armada EC2 dengan pembobotan instans	2012
Tutorial: Menggunakan Armada EC2 dengan Sesuai Permintaan sebagai kapasitas primer	2016
Tutorial: Meluncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas yang ditargetkan	2017
Tutorial: Meluncurkan instans ke Blok Kapasitas	2024
Tutorial: Menggunakan Armada Spot dengan pembobotan instans	2026
Contoh konfigurasi	2030
Contoh konfigurasi Armada EC2	2030
Konfigurasi contoh Armada Spot	2051
Kuota armada	2069
Meminta peningkatan kuota untuk kapasitas target	2071
Memantau	2072
Pemantauan otomatis dan manual	2073
Alat pemantauan otomatis	2074

Alat pemantauan manual	2075
Praktik terbaik untuk pemantauan	2076
Memantau status instans Anda	2077
Pemeriksaan status instans	2077
Peristiwa perubahan status	2086
Peristiwa terjadwal	2088
Pantau instans Anda menggunakan CloudWatch	2121
Alarm contoh	2122
Mengaktifkan pemantauan terperinci	2123
Membuat daftar metrik yang tersedia	2126
Instal dan konfigurasi CloudWatch agen	2152
Mendapatkan statistik untuk metrik	2156
Membuat grafik metrik	2166
Membuat alarm	2167
Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans	2168
Otomatisasi menggunakan EventBridge	2181
Tipe peristiwa Amazon EC2	2182
Log panggilan API menggunakan CloudTrail	2183
Informasi API Amazon EC2 di CloudTrail	2183
Memahami entri file log API Amazon EC2	795
Audit koneksi melalui EC2 Instance Connect	2186
Memantau aplikasi .NET dan SQL Server Anda	2187
Melacak penggunaan Tingkat Gratis Anda	2188
Jaringan	2192
Wilayah dan Zona	2193
Wilayah	2194
Zona Ketersediaan	2200
Zona Lokal	2205
Wavelength Zones	2207
AWS Outposts	2210
Pengalamatan IP instans	2212
Alamat IPv4 privat	2213
Alamat IPv4 publik	2214
Optimalisasi alamat IPv4 publik	2216
Alamat IP elastis (IPv4)	2217

Alamat IPv6	2217
Bekerja dengan alamat IPv4 untuk instans Anda	2219
Bekerja dengan alamat IPv6 untuk instans Anda	2222
Beberapa alamat IP	2225
Beberapa alamat IPv4 pribadi untuk Windows	2234
Nama host instans EC2	2242
Alamat link-lokal	2242
Tipe nama host instans	2243
Tipe nama host EC2	2243
Di mana Anda melihat Nama sumber daya dan nama IP	2245
Cara memutuskan apakah akan memilih nama Sumber Daya atau nama IP	2247
Modifikasi tipe Nama Host dan konfigurasi Nama host DNS	2247
Bawa alamat IP Anda sendiri	2249
Definisi BYOIP	2250
Persyaratan dan kuota	2251
Prasyarat orientasi	2252
Onboard BYOIP Anda	2260
Menggunakan rentang alamat Anda	2265
Validasi BYOIP Anda	2266
Ketersediaan wilayah	2270
Ketersediaan Local Zone	2270
Pelajari selengkapnya	2271
Alamat IP elastis	2271
Harga alamat IP Elastis	2272
Dasar alamat IP Elastis	2272
Cara menggunakan alamat IP Elastis	2273
Kuota alamat IP Elastis	2289
Antarmuka jaringan	2290
Dasar-dasar antarmuka jaringan	2291
Kartu jaringan	2293
Alamat IP per antarmuka jaringan per tipe instans	2295
Bekerja dengan antarmuka jaringan	2296
Praktik terbaik untuk mengonfigurasi antarmuka jaringan	2308
Skenario untuk antarmuka jaringan	2310
Antarmuka jaringan yang dikelola pemohon	2314
Tetapkan prefiks	2316

Bandwidth jaringan	2333
Bandwidth instans yang tersedia	2333
Memantau bandwidth instans	2335
Jaringan yang ditingkatkan	2336
Dukungan jaringan yang ditingkatkan	2336
Adaptor Jaringan Elastis (ENA)	2337
ENA Ekspres	2368
Intel 82599 VF	2391
Metrik performa jaringan	2404
Memecahkan masalah ENA di Linux	2414
Memecahkan masalah driver ENA Windows	2429
Meningkatkan latensi jaringan pada instans Linux	2449
Pertimbangan kinerja nitro	2453
Optimalkan kinerja jaringan pada instance Windows	2460
Elastic Fabric Adapter	2462
Dasar-dasar EFA	2463
Antarmuka dan pustaka yang didukung	2465
Tipe instans yang didukung	2465
Sistem operasi yang didukung	2466
Batasan EFA	2467
Harga EFA	2467
Memulai instans P5 dan EFA	2468
Memulai dengan EFA dan MPI	2472
Memulai dengan EFA dan NCCL	2488
Bekerja dengan EFA	2527
Memantau EFA	2531
Memverifikasi penginstal EFA menggunakan checksum	2531
Topologi instans	2543
Cara kerjanya	2544
Prasyarat	2548
Contoh	2550
Grup penempatan	2562
Strategi penempatan	2562
Aturan dan batasan	2566
Bekerja dengan grup penempatan	2569
Membagikan grup penempatan	2582

Grup penempatan di AWS Outposts	2588
MTU Network	2589
Frame jumbo (9001 MTU)	2590
Path MTU Discovery	2591
Periksa MTU jalur di antara dua host	2592
Periksa MTU untuk contoh Anda	2594
Tetapkan MTU untuk instans Anda	2595
Pemecahan Masalah	2598
Virtual private cloud	2598
VPC default Anda	2598
Membuat VPC tambahan	2599
Mengakses internet dari instans Anda	2600
Subnet bersama	2601
Subnet khusus IPv6	2601
Keamanan	2602
Perlindungan data	2603
Keamanan data Amazon EBS	2604
Enkripsi saat tidak aktif	2604
Enkripsi dalam transit	2606
Keamanan infrastruktur	2608
Isolasi jaringan	2608
Isolasi pada host fisik	2609
Mengontrol lalu lintas jaringan	2609
Ketahanan	2612
Validasi kepatuhan	2613
Manajemen identitas dan akses	2614
Akses jaringan ke instans Anda	2615
Atribut-atribut izin Amazon EC2	2615
IAM dan Amazon EC2	2616
Kebijakan IAM	2617
AWS kebijakan terkelola	2691
IAM role	2696
AWS PrivateLink	2713
Membuat titik akhir VPC antarmuka	2714
Membuat kebijakan titik akhir	2714
Manajemen pembaruan	2716

Praktik terbaik keamanan untuk instans Windows	2716
Praktik terbaik keamanan tingkat tinggi	2716
Manajemen pembaruan	2718
Manajemen konfigurasi	2720
Manajemen perubahan	2721
Audit dan akuntabilitas untuk instans Windows Amazon EC2	2722
Pasangan kunci	2722
Membuat pasangan kunci	2724
Menandai key pair	2733
Jelaskan pasangan kunci Anda	2735
Menghapus pasangan kunci Anda	2743
Menambahkan atau menghapus kunci publik pada instance Linux Anda	2745
Verifikasi sidik jari	2747
Grup keamanan	2749
Aturan-aturan grup keamanan	2751
Pelacakan koneksi	2754
Grup keamanan default dan kustom	2760
Cara menggunakan grup keamanan	2762
Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda	2772
NitroTPM	2780
Pertimbangan-pertimbangan	2781
Prasyarat	2781
Membuat AMI untuk dukungan NitroTPM	2783
Verifikasi apakah AMI sudah diaktifkan untuk NitroTPM	2784
Mengaktifkan atau menghentikan menggunakan NitroTPM pada instans	2785
Ambil kunci dukungan publik	2788
Credential Guard untuk instance Windows	2789
Prasyarat	2789
Luncurkan instance yang didukung	2790
Nonaktifkan integritas memori	2791
Aktifkan Credential Guard	2792
Verifikasi bahwa Credential Guard sedang berjalan	2794
Penyimpanan	2796
Amazon EBS	2797
Penyimpanan Instans	2798
Volume penyimpanan instans dan masa pakai data	2799

Volume penyimpanan instans	2802
Menambahkan volume penyimpanan instans	2804
Volume penyimpanan instans SSD	2810
Volume swap penyimpanan instans untuk instance Linux	2814
Optimalkan kinerja disk pada instance Linux	2818
Penyimpanan file	2820
Amazon S3	2820
Amazon EFS	2823
Amazon FSx	2826
Cache File Amazon	2832
Batasan volume instans	2832
Batas volume untuk instans yang dibangun di atas Sistem Nitro	2833
Batas volume untuk instans berbasis Xen	2835
Volume perangkat root	2837
Tipe volume root	2837
Pilih AMI Linux berdasarkan jenis volume root	2840
Tentukan jenis perangkat root dari instance Linux Anda	2841
Ubah volume root agar tetap ada	2842
Ubah ukuran awal volume root	2846
Mengganti volume root	2847
Nama perangkat	2859
Nama perangkat yang tersedia	2859
Pertimbangan nama perangkat	2861
Pemetaan perangkat blok	2863
Konsep pemetaan perangkat blok	2863
Pemetaan perangkat blok AMI	2867
Pemetaan perangkat blok instans	2871
Petakan disk ke volume	2879
Mencantumkan volume NVMe	2880
Mencantumkan volume	2885
Cuplikan Windows VSS EBS	2894
Apa itu VSS?	2895
Prasyarat	2897
Membuat snapshot VSS	2913
Memecahkan masalah snapshot EBS berbasis Windows VSS	2924
Pulihkan volume EBS dari snapshot EBS yang mendukung VSS	2928

Riwayat versi	2929
Pencegahan penulisan robek untuk instance Linux	2932
Harga	2933
Ukuran blok yang didukung dan perataan batas blok	2933
Persyaratan	2934
Periksa dukungan dan konfigurasi pencegahan tumpang tindih	2935
Konfigurasi perangkat lunak Anda untuk pencegahan tumpang tindih	2936
Sumber daya dan tanda	2939
Keranjang Sampah	2939
Bagaimana cara kerjanya?	2940
Sumber daya yang didukung	2941
Pertimbangan	2942
Kuota	2945
Layanan-layanan terkait	2945
Harga	2946
Izin IAM yang diperlukan	2946
Bekerja dengan aturan retensi	2951
Bekerja dengan sumber daya di Keranjang Sampah	2966
Pantau Keranjang Sampah	2976
Lokasi sumber daya	2995
ID sumber daya	2997
Membuat daftar dan memfilter sumber daya Anda	2998
Langkah-langkah konsol	2998
Langkah-langkah CLI dan API	3004
Tampilan Global (lintas Wilayah)	3007
Tampilan Global	3007
Tandai sumber daya Anda	3011
Dasar-dasar tanda	3011
Tandai sumber daya Anda	3013
Pembatasan tanda	3018
Manajemen tanda dan akses	3019
Menandai sumber daya Anda untuk penagihan	3019
Bekerja dengan tanda menggunakan konsol	3020
Bekerja dengan tanda menggunakan baris perintah	3026
Bekerja dengan tanda instans dalam metadata instans	3030
Tambahkan tag ke sumber daya menggunakan CloudFormation	3034

Kuota layanan	3035
Melihat kuota Anda saat ini	3035
Meminta peningkatan	3036
Pembatasan pada email yang dikirim menggunakan port 25	3037
Pecahkan Masalah	3038
Masalah umum dengan instans Windows	3038
Volume EBS tidak diinisialisasi di Windows Server 2016 dan 2019	3039
Lakukan boot instans Windows EC2 ke Directory Service Restore Mode (DSRM)	3040
Instans kehilangan konektivitas jaringan atau tugas terjadwal tidak berjalan saat diharapkan	3043
Tidak bisa mendapatkan output konsol	3044
Windows Server 2012 R2 tidak tersedia di jaringan	3044
Tabrakan tanda tangan disk	3044
Pesan umum dengan instance Windows	3046
“Kata sandi tidak tersedia”	3046
“Kata sandi belum tersedia”	3047
“Tidak dapat mengambil kata sandi Windows”	3048
“Menunggu layanan metadata”	3048
“Tidak dapat mengaktifkan Windows”	3052
“Windows tidak asli (0x80070005)”	3054
“Tidak ada Server Lisensi Server Terminal yang tersedia untuk memberikan lisensi”	3055
“Beberapa pengaturan dikelola oleh organisasi Anda”	3055
Pemecahan masalah peluncuran	3056
Nama perangkat tidak valid	3056
Batas instans terlampaui	3057
Kapasitas instans tidak cukup	3058
Konfigurasi yang diminta saat ini tidak didukung. Periksa dokumentasi untuk konfigurasi yang didukung.	3059
Instans langsung terhenti	3059
Izin tidak cukup	3061
Penggunaan CPU yang tinggi segera setelah Windows dimulai (hanya instance Windows)	3062
Hubungkan ke instans Linux Anda	3063
Penyebab umum masalah koneksi	3063
Kesalahan saat menghubungkan instans Anda: Waktu koneksi habis	3066
Kesalahan: tidak dapat memuat kunci ... Mengharapkan: KUNCI PRIVAT APA PUN	3069
Kesalahan: Kunci pengguna tidak dikenali oleh server	3070

Kesalahan: Izin ditolak atau koneksi ditutup oleh [instans] port 22	3072
Kesalahan: File kunci privat yang tidak dilindungi	3074
Kesalahan: Kunci privat harus dimulai dengan “-----BEGIN RSA PRIVATE KEY-----” dan diakhiri dengan “-----END RSA PRIVATE KEY-----”	3076
Kesalahan: Server menolak kunci kami atau Tidak tersedia metode autentikasi yang didukung	3076
Tidak dapat melakukan ping pada instans	3078
Kesalahan: Server menutup koneksi jaringan secara tidak terduga	3078
Kesalahan: Validasi kunci host gagal untuk EC2 Instance Connect	3078
Tidak dapat terhubung ke instans Ubuntu menggunakan EC2 Instance Connect	3080
Saya kehilangan kunci privat. Bagaimana caranya terhubung ke instans Linux saya?	3081
Hubungkan ke instans Windows Anda	3088
Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh	3089
Kesalahan menggunakan klien RDP macOS	3093
RDP menampilkan layar hitam, bukan desktop	3093
Tidak dapat masuk dari jarak jauh ke sebuah instans dengan pengguna yang bukan administrator	3094
Memecahkan masalah Remote Desktop menggunakan AWS Systems Manager	3094
Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh	3098
Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance Windows?	3099
Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa	3100
Atur ulang menggunakan EC2Launch v2	3101
Atur Ulang EC2Config	3106
Atur ulang menggunakan EC2Launch	3112
Memecahkan masalah instans yang tidak dapat dijangkau	3118
Boot ulang instans	3118
Output konsol instans	3119
Mengambil tangkapan layar instans yang tidak dapat dijangkau	3120
Tangkapan layar umum untuk instance Windows	3122
Pemulihan instans saat komputer host gagal	3132
Hentikan instans Anda	3132
Hentikan paksa instans	3133
Buat instans pengganti	3134
Akhiri instans Anda	3136
Instans langsung terhenti	3136
Penghentian instans yang tertunda	3136

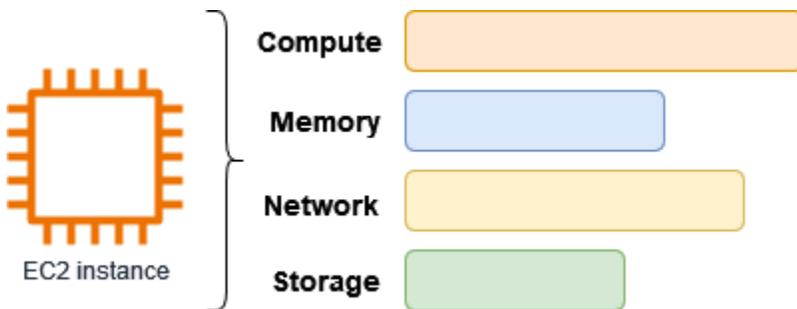
Instans yang dihentikan masih ditampilkan	3137
Kesalahan: Instans mungkin tidak dihentikan. Ubah atribut instance ApiTermination 'nonaktifkan'	3137
Instans diluncurkan atau dihentikan secara otomatis	3137
Pemeriksaan status gagal di Linux	3138
Meninjau informasi pemeriksaan status	3139
Mengambil log sistem	3140
Memecahkan masalah kesalahan log sistem untuk instance Linux	3140
Kehabisan memori: hentikan proses	3142
KESALAHAN: mmu_update gagal (Pembaruan manajemen memori gagal)	3143
Kesalahan I/O (kegagalan perangkat blok)	3143
KESALAHAN I/O: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi yang rusak)	3145
request_module: modprobe loop runaway (Melakukan loop modprobe kernel warisan pada versi Linux yang lebih lawas)	3146
“FATAL: kernel terlalu lawas” dan “fsck: Tidak ada file atau direktori tersebut saat mencoba membuka /dev” (Kernel dan AMI tidak cocok)	3148
“FATAL: Tidak dapat memuat /lib/modules” atau “BusyBox” (Modul kernel tidak ada)	3148
KESALAHAN Kernel tidak valid (kernel tidak kompatibel dengan EC2)	3150
fsck: Tidak ada file atau direktori tersebut saat mencoba membuka... (Sistem file tidak ditemukan)	3152
Kesalahan umum saat memasang sistem file (kegagalan pemasangan)	3154
VFS: Tidak dapat memasang fs root pada blok yang tidak diketahui (Sistem file root tidak cocok)	3156
Kesalahan: Tidak dapat menentukan jumlah perangkat root utama/minor... (Ketidakcocokan sistem file root/perangkat)	3158
XENBUS: Perangkat tanpa driver...	3159
... hari tanpa diperiksa, pemeriksaan paksa (Diperlukan pemeriksaan sistem file)	3160
fsck mati dengan status keluar... (Perangkat tidak ada)	3161
Perintah GRUB (grubdom>)	3162
Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat MAC dengan hard-coding)	3165
Tidak dapat memuat Kebijakan SELinux. Mesin berada dalam mode pemberlakuan. Menghentikan sekarang. (Kesalahan konfigurasi SELinux)	3167
XENBUS: Waktu menghubungkan ke perangkat habis (waktu habis Xenbus)	3168
Memecahkan masalah booting instance Linux dari volume yang salah	3169

Memecahkan masalah Sysprep	3171
EC2Rescue untuk Linux	3173
Menginstal EC2Rescue untuk Linux	3173
(Opsional) Verifikasi tanda tangan EC2Rescue untuk Linux	3175
Bekerja dengan EC2Rescue untuk Linux	3178
Mengembangkan modul EC2Rescue	3181
EC2Rescue untuk Windows Server	3187
Gunakan GUI	3188
Gunakan baris perintah	3195
Gunakan Systems Manager	3203
Konsol Serial EC2	3207
Prasyarat	3208
Konfigurasi akses ke Konsol Serial EC2	3216
Hubungkan ke Konsol Serial EC2	3225
Memutuskan koneksi dari Konsol Serial EC2	3233
Memecahkan masalah instans Anda menggunakan Konsol Serial EC2	3234
Kirimkan interupsi diagnostik	3244
Tipe instans yang didukung	3245
Prasyarat	3245
Kirimkan interupsi diagnostik	3249
Riwayat dokumen	3250
Sejarah untuk 2018 dan sebelumnya	3277
.....	mmmmcccv

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](#)

Otomatisasikan 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](#)

Otomatiskan pembuatan, pengelolaan, dan penyebaran gambar yang disesuaikan, aman, dan up-to-date server.

[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 end-to-end manajemen 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 kluster 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 AWS Management Console 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](#).

AWS CloudFormation

Amazon EC2 mendukung pembuatan sumber daya menggunakan AWS CloudFormation. Anda membuat template, dalam format JSON atau YAMAL, yang menjelaskan AWS sumber daya Anda, dan AWS CloudFormation ketentuan serta mengonfigurasi sumber daya tersebut untuk Anda. Anda dapat menggunakan kembali CloudFormation templat Anda untuk menyediakan sumber daya yang sama beberapa kali, baik di Wilayah dan akun yang sama atau di beberapa Wilayah 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](#).

AWS Tools for PowerShell

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

Instans Sesuai Permintaan

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 penggunaan Anda, baik On-Demand maupun sebagai bagian dari Savings Plans. Anda dapat menggunakan lisensi perangkat lunak terikat server yang ada dan mendapatkan bantuan untuk memenuhi persyaratan kepatuhan.

Reservasi Kapasitas Sesuai Permintaan

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

Penagihan per detik

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 Pricing Calculator](#).

[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 Billing and Cost Management](#). 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 Anda AWS Cost Explorer](#) di Panduan AWS Cost Management Pengguna.

Sumber daya

- [Fitur Amazon EC2](#)
- [AWS Re: posting](#)
- [AWS Pembangun Keterampilan](#)
- [AWS Support](#)
- [Tutorial Langsung](#)
- [Web Hosting](#)
- [Windows aktif AWS](#)

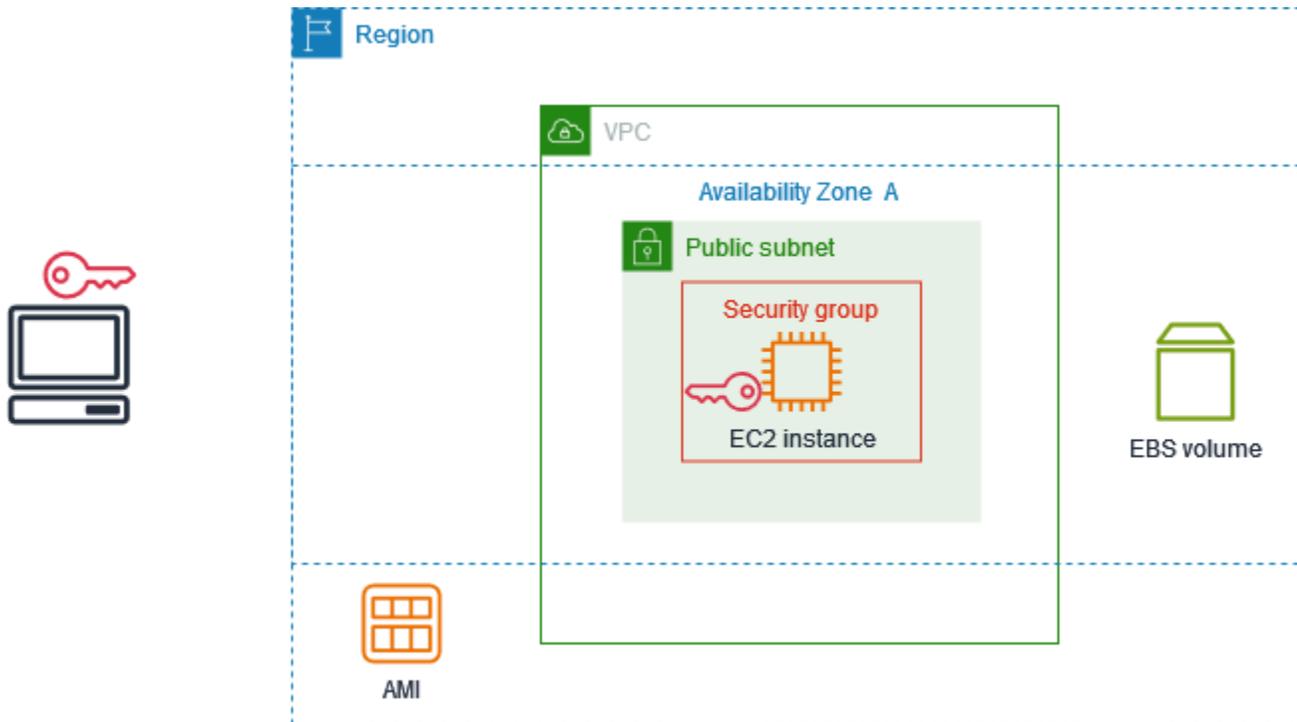
Mulai Amazon EC2

Gunakan tutorial ini untuk memulai Amazon Elastic Compute Cloud (Amazon EC2). Anda akan belajar cara meluncurkan dan terhubung ke instans EC2. 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.
- A key pair - Satu set kredensial keamanan yang Anda gunakan untuk membuktikan identitas Anda saat menghubungkan ke instans Anda. Kunci publik ada di instans Anda dan kunci pribadi ada di komputer Anda.
- 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 Availability Zone.
- Grup keamanan — Bertindak sebagai firewall virtual untuk mengontrol lalu lintas masuk dan keluar.
- Volume EBS — Kami memerlukan volume root untuk gambar. Anda dapat menambahkan volume data secara opsional.



Biaya untuk tutorial ini

Saat Anda mendaftar AWS, Anda dapat memulai dengan Amazon EC2 menggunakan file. [AWS Tingkat Gratis](#) Jika Anda membuat Akun AWS kurang dari 12 bulan yang lalu, dan 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. Sebaliknya, Anda akan dikenai biaya penggunaan Amazon EC2 standar sejak meluncurkan instans hingga mengakhiri instans (yang merupakan tugas akhir dari tutorial ini), meskipun instans tetap diam.

Untuk petunjuk untuk menentukan apakah Anda memenuhi syarat untuk Tingkat Gratis, lihat [the 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 instans EC2 menggunakan AWS Management Console seperti yang dijelaskan dalam prosedur berikut. Tutorial ini dimaksudkan untuk membantu Anda dengan cepat meluncurkan instance pertama Anda dalam manfaat Tingkat Gratis, sehingga 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, kami menampilkan arus Wilayah AWS - 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 Launch instance, pilih Launch 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 Cepat, lalu pilih sistem operasi (OS) untuk instans Anda. Untuk contoh Linux pertama Anda, kami sarankan Anda memilih Amazon Linux.
 - b. Dari Amazon Machine Image (AMI), pilih AMI yang ditandai Tingkat Gratis memenuhi syarat.
6. Di bawah Jenis instans, untuk jenis Instance `t2.micro`, pilih, mana yang memenuhi syarat untuk Tingkat Gratis. Di Wilayah `t2.micro` yang tidak `t3.micro` tersedia, memenuhi syarat untuk Tingkat Gratis.
7. Di bawah Key pair (login), untuk nama Key pair, pilih key pair yang ada atau pilih Create new key pair untuk membuat key pair pertama Anda.

Warning

Jika Anda memilih Proceed without a key pair (Tidak disarankan), Anda tidak akan dapat terhubung ke instance Anda menggunakan metode yang dijelaskan dalam tutorial ini.

8. Di bawah Pengaturan jaringan, perhatikan bahwa kami memilih VPC default Anda, memilih opsi untuk menggunakan subnet default di Availability Zone yang kami pilih untuk Anda, dan mengonfigurasi grup keamanan dengan aturan yang memungkinkan koneksi ke instans Anda dari mana saja. 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 dan kemudian 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 Anda.
 - (Opsional) Untuk membatasi lalu lintas koneksi masuk ke jaringan tertentu, pilih Kustom alih-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 memungkinkan lalu lintas koneksi dari jaringan Anda, Anda tidak akan dapat terhubung ke instans Anda. Untuk contoh Linux, Anda harus mengizinkan lalu lintas SSH. Untuk contoh Windows, Anda harus mengizinkan lalu lintas RDP.
9. Di bawah Konfigurasi penyimpanan, perhatikan bahwa kami mengonfigurasi volume root 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 contoh. Keadaan instance 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 menghubungkan ke instans 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 SSH client.
5. (Opsional) Jika Anda membuat key pair saat meluncurkan instance dan mengunduh kunci pribadi (file.pem) ke komputer yang menjalankan Linux atau macOS, jalankan chmod perintah example 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 14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

8. (Opsional) Verifikasi bahwa sidik jari dalam peringatan keamanan cocok dengan sidik jari instance yang terdapat dalam keluaran konsol saat Anda pertama kali memulai instance. Untuk mendapatkan output konsol, pilih Actions, Monitor dan troubleshoot, Dapatkan log sistem. Jika sidik jari tidak cocok, seseorang mungkin mencoba menyerang. 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 terhubung ke instance Windows, Anda harus mengambil kata sandi administrator awal dan menggunakan kata sandi ini saat Anda terhubung ke instans Anda menggunakan Remote Desktop. Diperlukan beberapa menit setelah peluncuran instans sebelum sandi ini tersedia.

Nama pengguna default untuk akun Administrator tergantung pada bahasa sistem operasi (OS) yang terkandung dalam AMI. Untuk memastikan nama pengguna yang benar, identifikasi bahasa OS AMI Anda, lalu pilih nama pengguna yang sesuai. Misalnya, untuk OS bahasa Inggris, nama pengguna 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 pengguna `Administrator` (`Other`). Untuk informasi selengkapnya, lihat [Nama Lokal untuk Akun Administrator di Windows](#) di Microsoft TechNet Wiki.

Jika Anda telah menggabungkan instans Anda ke suatu domain, Anda dapat ter-connect ke instans Anda menggunakan kredensial domain yang telah Anda tentukan di AWS Directory Service. Pada layar login Remote Desktop, alih-alih menggunakan nama komputer lokal dan kata sandi yang dibuat, gunakan nama pengguna yang memenuhi syarat untuk administrator (misalnya `corp.example.com \Admin`) dan kata sandi untuk akun ini.

Jika Anda menerima kesalahan saat mencoba untuk terhubung ke instance Anda, lihat [the section called “Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh”](#).

Untuk terhubung ke instans Windows menggunakan klien RDP

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 instance 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 kunci pribadi (.pem) yang Anda tentukan saat meluncurkan instance. Pilih file dan pilih Buka untuk menyalin seluruh isi file ke jendela ini.
 - b. Pilih Dekripsi kata sandi. Halaman Dapatkan kata sandi Windows ditutup, dan kata sandi administrator default untuk instance muncul di bawah Kata Sandi, 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.
8. Pilih Unduh file desktop jarak jauh. Setelah selesai mengunduh file, pilih Batalkan untuk kembali ke halaman Instans. Arahkan ke direktori unduhan Anda, dan buka file RDP.
 9. Anda mungkin mendapatkan peringatan bahwa penerbit koneksi jarak jauh tidak dikenal. Pilih Hubungkan untuk terus terhubung ke instans Anda.
 10. Akun administrator dipilih secara default. Tempel kata sandi yang Anda salin sebelumnya, lalu pilih OK.
 11. Karena sifat dari sertifikat yang ditandatangani sendiri, Anda mungkin mendapatkan peringatan bahwa sertifikat keamanan tidak dapat diautentikasi. Lakukan salah satu hal berikut ini:
 - Jika Anda mempercayai sertifikat, pilih Ya untuk terhubung ke instans Anda.
 - [Windows] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai dalam 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 Actions, Monitor dan troubleshoot, Dapatkan log sistem.
 - [Mac OS X] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai dalam log sistem untuk mengonfirmasi identitas komputer jarak jauh. Pilih Tampilkan Sertifikat, perluas Detail, dan pilih Sidik Jari SHA1. Bandingkan nilai ini dengan nilai RDPCERTIFICATE-THUMBPRINT di Actions, Monitor dan troubleshoot, Dapatkan log sistem.

Langkah 3: Bersihkan instans Anda

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](#).

⚠ Important

Mengakhiri sebuah instans secara efektif akan menghapusnya; Anda tidak dapat terhubung kembali ke sebuah instans setelah mengakhirinya.

Jika Anda meluncurkan sebuah instans yang tidak berada dalam [AWS Tingkat Gratis](#), Anda tidak akan lagi dikenai biaya untuk instans tersebut setelah status instans berubah menjadi `shutting down` atau `terminated`. Untuk menjaga instans agar tetap dalam versi yang terbaru, tetapi tidak dikenai biaya, Anda dapat menghentikan instans sekarang dan kemudian mulai lagi nanti. Untuk informasi selengkapnya, lihat [Hentikan dan mulai instans Amazon EC2](#).

Untuk mengakhiri instans Anda

1. Di panel navigasi, pilih Instans. Dalam daftar instans, pilih instans tersebut.
2. Pilih Status instans, Akhiri instans.
3. Pilih Akhiri saat diminta untuk mengonfirmasi.

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 instans, Anda mungkin ingin menjelajahi langkah-langkah berikut:

- Pelajari cara melacak penggunaan Tingkat Gratis Anda untuk menghindari kejutan penagihan. Untuk informasi selengkapnya, lihat [the section called “Melacak penggunaan Tingkat Gratis Anda”](#).
- Konfigurasi CloudWatch alarm untuk memberi tahu Anda jika penggunaan Anda melebihi Tingkat Gratis. Untuk informasi selengkapnya, lihat [Melacak penggunaan Tingkat AWS Gratis 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 pembelian instans](#).
- Dapatkan saran tentang tipe instans. Untuk informasi selengkapnya, lihat [Dapatkan rekomendasi tipe instans untuk beban kerja baru](#).

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 API menggunakan federasi identitas dengan penyedia identitas dan peran IAM bila memungkinkan. Untuk informasi selengkapnya, lihat [Membuat kebijakan IAM](#) dalam Panduan Pengguna IAM.
- Terapkan aturan paling sedikit permisif untuk grup keamanan Anda. Untuk informasi selengkapnya, lihat [Aturan-aturan grup keamanan](#).
- 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, lihat [Praktik 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 kontrol untuk memantau sumber daya Amazon EC2 Anda terhadap praktik terbaik keamanan dan standar keamanan. Untuk informasi selengkapnya tentang penggunaan Security Hub, lihat [kontrol Amazon Elastic Compute Cloud](#) dalam Panduan Pengguna AWS Security Hub .

Penyimpanan

- Memahami implikasi tipe perangkat root untuk persistensi, pencadangan, dan pemulihan data. Untuk informasi selengkapnya, lihat [Penyimpanan untuk perangkat 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 kluster 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 [Cara menggunakan metadata instans](#) 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 Support .

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 Backup](#) dan [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 [Memantau 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.

Amazon Machine Image (AMI)

Amazon Machine Image (AMI) adalah gambar yang didukung dan dipelihara AWS yang disediakan oleh yang menyediakan informasi yang diperlukan untuk meluncurkan instance. Anda harus menentukan AMI saat meluncurkan instans. Anda dapat meluncurkan beberapa instans dari satu AMI jika Anda memerlukan beberapa instans dengan konfigurasi yang sama. Anda dapat menggunakan AMI berbeda untuk meluncurkan instans jika Anda memerlukan konfigurasi yang berbeda.

AMI mencakup yang berikut ini:

- Satu atau beberapa snapshot Amazon Elastic Block Store (Amazon EBS), atau, instance-store-backed untuk AMI, template untuk volume root instance (misalnya, sistem operasi, server aplikasi, dan aplikasi).
- Luncurkan izin yang mengontrol AWS akun mana yang dapat menggunakan AMI untuk meluncurkan instance.
- Pemetaan perangkat blok yang menentukan volume yang ditambahkan ke instans saat diluncurkan.

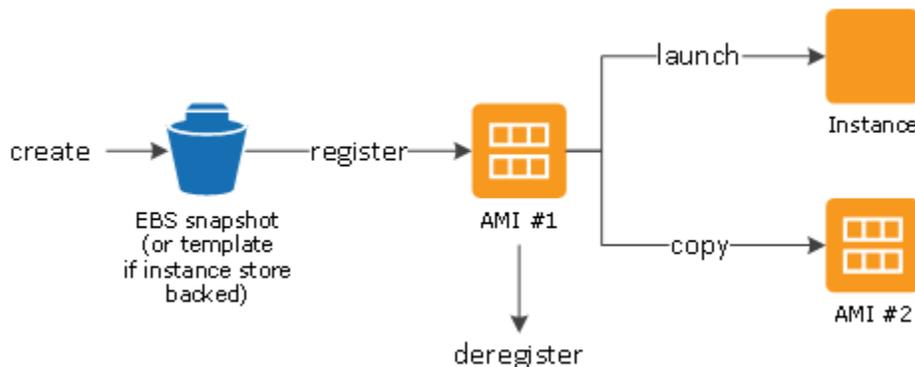
Topik Amazon Machine Image (AMI)

- [Menggunakan AMI](#)
- [Membuat AMI Anda sendiri](#)
- [Membeli, berbagi, dan menjual AMI](#)
- [Membatalkan pendaftaran AMI Anda](#)
- [Amazon Linux 2023 dan Amazon Linux 2](#)
- [AMI Windows](#)
- [Jenis AMI](#)
- [Tipe virtualisasi AMI](#)
- [Mode boot Amazon EC2](#)
- [Temukan AMI](#)
- [AMI bersama](#)
- [AMI berbayar](#)
- [Siklus hidup AMI](#)
- [Menggunakan enkripsi dengan AMI yang didukung EBS](#)

- [Pantau peristiwa AMI menggunakan Amazon EventBridge](#)
- [Memahami informasi penagihan AMI](#)
- [Kuota AMI](#)

Menggunakan AMI

Diagram berikut merangkum siklus hidup AMI. Setelah membuat dan mendaftarkan AMI, Anda dapat menggunakannya untuk meluncurkan instans baru. (Anda juga dapat meluncurkan instans dari AMI jika pemilik AMI memberi Anda izin peluncuran.) Anda dapat menyalin AMI dalam Wilayah yang sama atau ke AWS AWS Wilayah yang berbeda. Jika Anda tidak lagi memerlukan AMI, Anda dapat membatalkan pendaftarannya.



Anda dapat mencari AMI yang memenuhi kriteria untuk instans Anda. Anda dapat mencari AMI yang disediakan oleh AWS atau AMI yang disediakan oleh komunitas. Untuk informasi lebih lanjut, lihat [Jenis AMI](#) dan [Temukan AMI](#).

Setelah Anda meluncurkan instans dari suatu AMI, Anda dapat membuat sambungan ke instans tersebut. Saat Anda terhubung ke sebuah instans, Anda dapat menggunakan instans tersebut seperti Anda menggunakan server lain. Untuk informasi tentang peluncuran, melakukan sambungan, dan menggunakan instans Anda, lihat [Mulai Amazon EC2](#).

Membuat AMI Anda sendiri

Anda dapat meluncurkan instance dari AMI yang ada, menyesuaikan instance (misalnya, [menginstal perangkat lunak](#) pada instance), dan kemudian menyimpan konfigurasi yang diperbarui ini sebagai AMI kustom. Instans yang diluncurkan dari AMI kustom tersebut sudah mencakup kustomisasi yang Anda buat saat membuat AMI.

Perangkat penyimpanan root instans akan menentukan proses yang harus Anda lakukan untuk membuat AMI. Volume root instans adalah volume Amazon Elastic Block Store (Amazon EBS) atau volume penyimpanan instans. Untuk informasi selengkapnya tentang volume perangkat root, lihat [Volume root instans Amazon EC2](#).

- Untuk membuat AMI yang didukung Amazon EBS, lihat [Buat AMI yang didukung Amazon EBS-Backed](#).
- Untuk membuat AMI yang didukung penyimpanan instans, lihat [Buat AMI Linux yang didukung penyimpanan instans](#).

Untuk membantu mengelompokkan dan mengelola AMI, Anda dapat menetapkan tag kustom pada AMI tersebut. Untuk informasi selengkapnya, lihat [Tandai sumber daya Amazon EC2 Anda](#).

Membeli, berbagi, dan menjual AMI

Setelah Anda membuat AMI, Anda dapat merahasiakannya sehingga hanya Anda yang dapat menggunakannya, atau Anda dapat membagikannya dengan daftar AWS akun tertentu. Anda juga dapat membuat AMI kustom Anda publik sehingga komunitas dapat menggunakannya. 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 [AMI bersama](#).

Anda dapat membeli AMI dari pihak ketiga, termasuk AMI yang disertakan dalam kontrak layanan dari organisasi, seperti Red Hat. Anda juga dapat membuat AMI dan menjualnya ke pengguna Amazon EC2 lainnya. Untuk informasi selengkapnya tentang pembelian atau penjualan AMI, lihat [AMI berbayar](#).

Membatalkan pendaftaran AMI Anda

Anda dapat membatalkan pendaftaran AMI setelah selesai digunakan. Setelah Anda membatalkan pendaftaran AMI, maka AMI tersebut tidak dapat digunakan untuk meluncurkan instans baru. Instans yang ada yang telah diluncurkan dari AMI tersebut tidak terpengaruh. Untuk informasi selengkapnya, lihat [Deregister \(hapus\) AMI](#).

Amazon Linux 2023 dan Amazon Linux 2

Rilis terbaru Amazon Linux, AL2023, dioptimalkan untuk Amazon EC2 dan tersedia tanpa biaya tambahan untuk para pengguna Amazon EC2. Fitur AL2023 mencakup cadence dengan peluncuran yang dapat diprediksi, pembaruan frekuen, dan dukungan jangka panjang.

Untuk informasi selengkapnya tentang fitur AL2023 dan meluncurkan AMI AL2023, lihat:

- [Fitur-Fitur AL2023](#)
- [Mulai AL2023](#)

Amazon Linux 2 (AL2) menyediakan lingkungan eksekusi yang stabil, aman, dan berkinerja tinggi untuk aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya tentang Amazon Linux 2, lihat [Amazon Linux 2 di Amazon EC2](#) di Panduan Pengguna Amazon Linux 2.

Note

Amazon Linux AMI mencapai puncaknya end-of-life pada 31 Desember 2023 dan tidak akan menerima pembaruan keamanan atau perbaikan bug mulai 1 Januari 2024. Untuk informasi selengkapnya tentang AMI Amazon Linux end-of-life dan dukungan pemeliharaan, lihat posting blog [Pembaruan di Amazon Linux AMI end-of-life](#). Sebaiknya Anda memutakhirkan aplikasi ke AL2023, yang mencakup dukungan jangka panjang hingga 2028.

AMI Windows

AWS menyediakan satu set AMI yang tersedia untuk umum yang berisi konfigurasi perangkat lunak khusus untuk platform Windows. Anda dapat mulai cepat membangun dan melakukan deploy aplikasi Anda dengan Amazon EC2 menggunakan AMI ini. Pertama, pilih AMI yang sesuai kebutuhan spesifik Anda, kemudian luncurkan instans menggunakan AMI tersebut. Anda mengambil kata sandi untuk akun administrator, lalu masuk ke instans menggunakan Koneksi Desktop Jarak Jauh, sama seperti Anda menggunakan Windows Server lainnya. Untuk informasi selengkapnya tentang AMI AWS Windows, lihat [referensi AWS Windows AMI](#).

Saat Anda meluncurkan instans dari AMI Windows, perangkat root untuk instans Windows adalah volume Amazon Elastic Block Store (Amazon EBS). AMI Windows tidak mendukung penyimpanan instance untuk perangkat root.

AMI Windows yang dikonfigurasi untuk peluncuran lebih cepat dengan EC2 Fast Launch telah disediakan sebelumnya, menggunakan snapshot untuk meluncurkan instans hingga 65% lebih cepat. Untuk mempelajari selengkapnya tentang Peluncuran Cepat EC2, lihat [Gunakan EC2 Fast Launch untuk instans Windows Anda](#).

Note

Microsoft tidak lagi mendukung versi Windows Server sebelum Windows Server 2016. Kami menyarankan Anda meluncurkan instans EC2 baru menggunakan versi Windows Server yang didukung. Jika Anda memiliki instans EC2 yang menjalankan versi Windows Server yang tidak didukung, kami menyarankan Anda untuk meningkatkan instans tersebut ke versi Windows Server yang didukung. Untuk informasi selengkapnya, lihat [Mutakhirkan instans Amazon EC2 Windows ke versi Windows Server yang lebih baru](#).

Jenis AMI

Anda dapat memilih AMI untuk digunakan berdasarkan karakteristik berikut:

- Wilayah (lihat [Wilayah dan Zona](#))
- Sistem operasi
- Arsitektur (32-bit atau 64-bit)
- [Izin peluncuran](#)
- [Penyimpanan untuk perangkat root](#)

Izin peluncuran

Pemilik AMI 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.

Izin peluncuran	Deskripsi
implisit	Pemilik memiliki izin peluncuran implisit untuk AMI.

Amazon dan komunitas Amazon EC2 menyediakan banyak pilihan AMI publik. Untuk informasi selengkapnya, lihat [AMI bersama](#). Developer dapat membebaskan biaya untuk AMI yang mereka buat. Untuk informasi selengkapnya, lihat [AMI berbayar](#).

Penyimpanan untuk perangkat root

Semua AMI dikategorikan sebagai didukung oleh Amazon EBS atau didukung oleh penyimpanan instans.

- AMI yang didukung Amazon EBS – Perangkat root untuk instans yang diluncurkan dari AMI adalah volume Amazon Elastic Block Store (Amazon EBS) yang dibuat dari snapshot Amazon EBS. Didukung untuk Linux dan Windows AMI.
- AMI yang didukung penyimpanan instans Amazon – Perangkat root untuk instans yang diluncurkan dari AMI adalah volume penyimpanan instans yang dibuat dari templat yang tersimpan di Amazon S3. Didukung hanya untuk AMI Linux. AMI Windows tidak mendukung penyimpanan instans untuk perangkat root.

Untuk informasi selengkapnya, lihat [Volume root instans Amazon EC2](#).

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

Karakteristik	AMI yang didukung Amazon EBS	AMI yang didukung oleh penyimpanan instans Amazon
Waktu boot untuk instans	Biasanya kurang dari 1 menit	Biasanya kurang dari 5 menit
Batas ukuran untuk perangkat root	64 TiB**	10 GiB
Volume perangkat root	Volume EBS	Volume penyimpanan instans

Karakteristik	AMI yang didukung Amazon EBS	AMI yang didukung oleh penyimpanan instans Amazon
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.
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, penggunaan volume EBS, dan penyimpanan AMI sebagai snapshot EBS.	Anda dikenai biaya penggunaan instans dan penyimpanan AMI di Amazon S3.
Pembuatan/pemaketa n AMI	Menggunakan perintah/panggilan tunggal	Memerlukan instalasi dan penggunaan alat AMI
Kondisi terhenti	Dapat berada dalam kondisi terhenti. Bahkan saat instans terhenti dan tidak berjalan, volume root tetap ada di Amazon EBS	Tidak dapat dalam kondisi terhenti; instans berjalan atau diakhiri

* 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 [Ubah volume root agar tetap ada](#).

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

Tentukan tipe perangkat root AMI Anda

Untuk menentukan tipe perangkat root AMI menggunakan konsol

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` – Ini adalah AMI yang didukung EBS.
 - `instance store` – Ini adalah AMI yang didukung penyimpanan instans.

Untuk menentukan tipe perangkat root sebuah AMI menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-images](#) (AWS CLI)
- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

Kondisi terhenti

Anda dapat menghentikan instans yang memiliki volume EBS sebagai perangkat root-nya, tetapi Anda tidak dapat menghentikan instans yang memiliki volume penyimpanan instans sebagai perangkat root-nya.

Penghentian menyebabkan instans berhenti berjalan (statusnya berubah dari `running` ke `stopping` ke `stopped`). Instans yang dihentikan tetap ada di Amazon EBS sehingga dapat dimulai ulang. Penghentian berbeda dari pengakhiran; Anda tidak dapat memulai ulang instans yang diakhiri. Karena instans dengan volume penyimpanan instans sebagai perangkat root-nya tidak dapat dihentikan, instans tersebut berjalan atau diakhiri. Untuk informasi selengkapnya tentang apa yang terjadi dan apa yang dapat Anda lakukan saat sebuah instans dihentikan, lihat [Hentikan dan mulai instans Amazon EC2](#).

Penyimpanan dan persistensi data default

Instans yang memiliki volume penyimpanan instans sebagai perangkat root-nya akan secara otomatis memiliki penyimpanan instans (volume root berisi partisi root dan Anda dapat menyimpan data tambahan). Anda dapat menambahkan penyimpanan tetap ke instans dengan melampirkan satu

atau beberapa volume Amazon EBS. Data apa pun pada volume penyimpanan instans akan dihapus ketika instans gagal atau berakhir. Untuk informasi selengkapnya, lihat [Volume penyimpanan instans dan masa pakai data](#).

Instans yang memiliki Amazon EBS sebagai perangkat root-nya akan secara otomatis dilampirkan volume EBS. Volume ini muncul dalam daftar volume Anda seperti yang lainnya. Untuk sebagian besar tipe instans, instans yang memiliki volume EBS sebagai perangkat root-nya tidak memiliki volume penyimpanan instans secara default. Anda dapat menambahkan volume penyimpanan instans atau volume EBS tambahan dengan menggunakan pemetaan perangkat blok. Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#).

Waktu boot

Instans yang diluncurkan dari peluncuran AMI yang didukung Amazon EBS akan diluncurkan lebih cepat daripada instans yang diluncurkan dari AMI yang didukung penyimpanan instans. Saat Anda meluncurkan instans dari AMI yang didukung penyimpanan instans, semua bagian harus diambil dari Amazon S3 sebelum instans tersedia. Untuk AMI yang didukung Amazon EBS, hanya bagian yang diperlukan untuk melakukan boot instans yang perlu diambil dari snapshot sebelum instans tersebut tersedia. Namun, performa instans yang menggunakan volume EBS sebagai perangkat root-nya akan lebih lambat untuk waktu singkat ketika bagian lainnya sedang diambil dari snapshot dan dimuat ke dalam volume. Saat Anda menghentikan dan memulai ulang instans, instans akan diluncurkan dengan cepat karena kondisi tersebut disimpan dalam volume EBS.

Pembuatan AMI

Untuk membuat AMI Linux yang didukung oleh penyimpanan instans, Anda harus membuat AMI dari instans milik Anda pada instans itu sendiri menggunakan alat Amazon EC2 AMI. Perhatikan bahwa AMI Windows tidak mendukung penyimpanan instance untuk perangkat root.

Pembuatan AMI jauh lebih mudah untuk AMI yang didukung oleh Amazon EBS. Tindakan API `CreateImage` akan membuat AMI yang didukung Amazon EBS dan mendaftarkannya. Ada juga tombol di dalamnya AWS Management Console yang memungkinkan Anda membuat AMI dari instance yang sedang berjalan. Untuk informasi selengkapnya, lihat [Buat AMI yang didukung Amazon EBS-Backed](#).

Bagaimana Anda dikenai biaya

Untuk AMI yang didukung oleh penyimpanan instans, Anda dikenai biaya untuk penggunaan instans dan penyimpanan AMI Anda di Amazon S3. Untuk AMI yang didukung oleh Amazon EBS,

Anda dikenai biaya untuk penggunaan instans, penyimpanan dan penggunaan volume EBS, serta penyimpanan AMI sebagai snapshot EBS.

Untuk AMI yang didukung penyimpanan instans Amazon EC2, setiap kali Anda mengustomisasi AMI dan membuat yang baru, semua bagian disimpan di Amazon S3 untuk setiap AMI. Jadi, jejak penyimpanan untuk setiap AMI kustom adalah sebesar AMI tersebut. Untuk AMI yang didukung Amazon EBS, setiap kali Anda mengustomisasi AMI dan membuat yang baru, hanya perubahan yang disimpan. Jadi, jejak penyimpanan untuk AMI selanjutnya yang Anda kustomisasikan setelahnya akan lebih kecil, sehingga Anda dikenai biaya penyimpanan AMI yang lebih rendah.

Saat instans dengan volume EBS sebagai perangkat root-nya dihentikan, Anda tidak dikenai biaya untuk penggunaan instans, tetapi Anda masih dikenai biaya untuk penyimpanan volume. Segera setelah Anda memulai instans, kami akan membebaskan biaya penggunaan minimal satu menit. Setelah satu menit, kami membebaskan biaya hanya untuk detik yang digunakan. Misalnya, jika Anda menjalankan instans selama 20 detik, lalu menghentikannya, kami akan membebaskan biaya selama satu menit penuh. Jika Anda menjalankan instans selama 3 menit 40 detik, kami membebaskan biaya tepat 3 menit dan 40 detik penggunaan. Kami membebaskan biaya untuk setiap detik, dengan minimum satu menit, untuk menjaga agar instans tetap berjalan meski instans tersebut tidak aktif dan Anda tidak tersambung ke instans tersebut.

Tipe virtualisasi AMI

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.

Untuk performa terbaik, kami sarankan Anda menggunakan tipe instans generasi saat ini dan AMI HVM saat Anda meluncurkan instans. Untuk informasi selengkapnya tentang tipe instans generasi saat ini, lihat [Tipe Instans Amazon EC2](#). Jika Anda menggunakan tipe instans generasi sebelumnya dan ingin menggunakan yang baru, lihat [Jalur Peningkatan](#) dan [Ubah tipe instans](#).

Tabel berikut ini membandingkan AMI HVM dan PV.

	HVM	PV
Deskripsi	AMI HVM disajikan dengan set perangkat keras virtual	AMI PV melakukan boot dengan bootloader khusus

	HVM	PV
	<p>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.</p>	<p>yang disebut PV-GRUB, yang memulai siklus boot, lalu memuat secara berantai kernel yang ditentukan dalam file menu.lst pada gambar Anda. Tamu Paravirtual dapat menjalankan perangkat keras host yang tidak memiliki dukungan virtualisasi secara eksplisit. Secara historis, tamu PV biasanya memiliki performa yang lebih baik dibandingkan tamu HVM, tetapi karena peningkatan virtualisasi HVM dan ketersediaan driver PV untuk HVM HMI, ini tidak lagi benar. Untuk informasi selengkapnya tentang PV-GRUB dan penggunaannya di Amazon EC2, lihat Kernel yang disediakan pengguna.</p>

	HVM	PV
Dukungan untuk ekstensi perangkat keras	<p>Ya. Tidak seperti tamu PV, tamu HVM dapat memanfaatkan ekstensi perangkat keras yang menyediakan akses cepat ke perangkat keras dasar di sistem host. Untuk informasi selengkapnya tentang ekstensi virtualisasi CPU yang tersedia di Amazon EC2, lihat Teknologi Virtualisasi Intel di situs web Intel.</p> <p>AMI HVM diperlukan untuk memanfaatkan peningkatan jaringan dan pemrosesan GPU. Agar dapat meneruskan instruksi ke perangkat jaringan dan GPU khusus, OS perlu memiliki akses ke platform perangkat keras asli; virtualisasi HVM menyediakan akses ini. Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di Amazon EC2.</p>	<p>Tidak, keduanya tidak dapat memanfaatkan ekstensi perangkat keras khusus, seperti peningkatan jaringan atau pemrosesan GPU.</p>
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.

	HVM	PV
Wilayah yang didukung	Semua Wilayah mendukung instans HVM.	Asia Pasifik (Tokyo), Asia Pasifik (Singapura), Asia Pasifik (Sydney), Eropa (Frankfurt), Eropa (Irlandia), Amerika Selatan (Sao Paulo), AS Timur (Virginia Utara), AS Barat (California Utara), dan As Barat (Oregon)
Cara menemukan	Verifikasikan bahwa tipe virtualisasi AMI diatur ke hvm, menggunakan konsol atau perintah describe-images . Untuk informasi selengkapnya, lihat Temukan AMI .	Verifikasikan bahwa tipe virtualisasi AMI diatur ke paravirtual , menggunakan konsol atau perintah describe-images . Untuk informasi selengkapnya, lihat Temukan AMI .

PV di HVM

Tamu paravirtual pada umumnya memiliki performa lebih baik dengan operasi penyimpanan dan jaringan daripada tamu HVM karena dapat memanfaatkan driver khusus untuk I/O yang menghindari overhead emulasi perangkat keras jaringan dan disk, sedangkan tamu HVM harus menerjemahkan instruksi ini ke perangkat keras yang diemulasi. Sekarang, driver PV tersedia untuk tamu HVM, sehingga sistem operasi yang tidak dapat dijalankan di lingkungan paravirtual masih dapat merasakan keunggulan performa dalam penyimpanan dan I/O jaringan dengan menggunakannya. Dengan PV pada driver HVM ini, tamu HVM dapat memperoleh performa yang sama, atau lebih baik daripada tamu paravirtual.

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 AMI `uefi-preferred` dengan tipe instans yang tidak mendukung UEFI, instans akan diluncurkan sebagai Legacy BIOS dan fitur yang bergantung pada UEFI akan dinonaktifkan. Jika Anda mengandalkan ketersediaan fitur yang bergantung pada UEFI, atur parameter mode boot AMI Anda ke `uefi`.

Mode boot default sesuai tipe instans

- Tipe instans Graviton: UEFI
- Tipe instans Intel dan AMD: Legacy BIOS

Menjalankan tipe instans Intel dan AMD di UEFI

[Most Intel and AMD instance types](#) dapat berjalan di UEFI dan Legacy BIOS. Untuk menggunakan UEFI, Anda harus memilih AMI dengan parameter mode boot `uefi` atau `uefi-preferred`, dan sistem operasi yang berada dalam AMI harus dikonfigurasi untuk mendukung UEFI.

Topik mode boot

- [Meluncurkan instans](#)
- [Tentukan parameter mode boot suatu AMI](#)
- [Menentukan mode boot yang didukung untuk sebuah tipe instans](#)
- [Menentukan mode boot dari sebuah instans](#)
- [Menentukan mode boot sistem operasi](#)
- [Mengatur mode boot sebuah AMI](#)
- [Variabel UEFI](#)
- [UEFI Secure Boot](#)

Meluncurkan instans

Anda dapat meluncurkan instans dalam mode boot UEFI atau Legacy BIOS.

Topik

- [Batasan](#)
- [Pertimbangan](#)
- [Persyaratan untuk meluncurkan sebuah instans di UEFI](#)

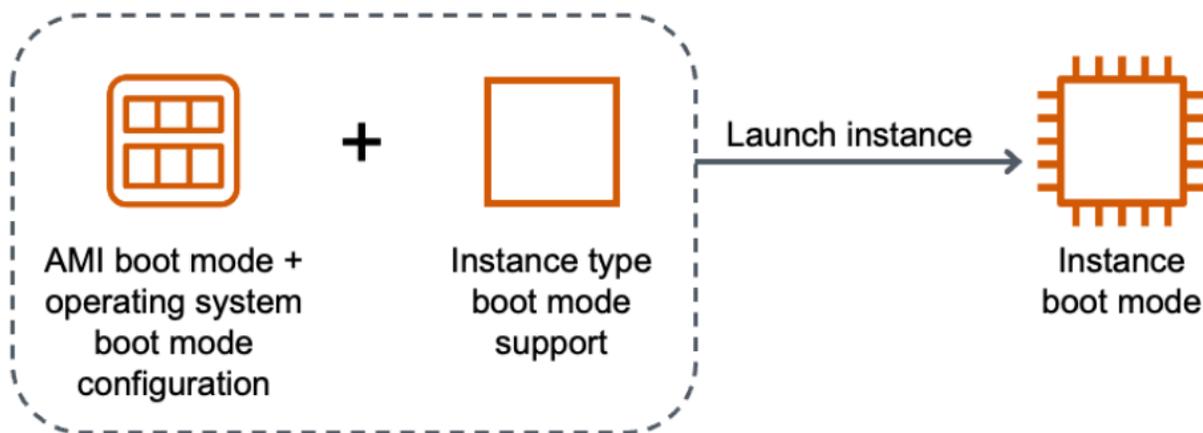
Batasan

Boot UEFI tidak didukung di Zona Lokal, Zona Wavelength, atau dengan AWS Outposts.

Pertimbangan

Pertimbangkan hal-hal berikut saat meluncurkan instans:

- Mode boot instans ditentukan oleh konfigurasi AMI, sistem operasi yang berada di dalamnya, dan tipe instans, diilustrasikan oleh gambar berikut:



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
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

Parameter mode boot AMI	Konfigurasi mode boot sistem operasi	Dukungan mode boot tipe instans	Mode boot instans yang dihasilkan
Tidak ada mode boot yang ditentukan - x86	Legacy BIOS	UEFI dan Legacy BIOS	Legacy BIOS

- Mode boot default:
 - Tipe instans Graviton: UEFI
 - Tipe instans Intel dan AMD: Legacy BIOS
- Tipe instans Intel dan AMD yang mendukung UEFI, selain Legacy BIOS:
 - Semua instans dibangun pada Sistem AWS Nitro, kecuali: instans logam kosong, DL1, G4ad, P4, u-3tb1, u-6tb1, u-9tb1, u-12tb1, u-18tb1, u-24tb1, dan VT1

Untuk melihat jenis instans yang tersedia yang mendukung UEFI di Wilayah tertentu

Tipe instans yang tersedia berbeda-beda menurut Wilayah AWS. Untuk melihat tipe instans yang tersedia yang mendukung UEFI di suatu Wilayah, gunakan perintah [describe-instance-types](#) dengan parameter `--region`. Jika Anda menghilangkan parameter `--region`, [Wilayah default](#) Anda akan digunakan dalam permintaan. Sertakan parameter `--filters` untuk cakupan hasil ke tipe instans yang mendukung UEFI dan parameter `--query` untuk cakupan output ke nilai InstanceType.

Gunakan perintah untuk sistem operasi Anda.

Linux

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge
c5.12xlarge
```

...

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object {$_.SupportedBootModes -Contains "uefi"} | `
  Sort-Object InstanceType | `
  Format-Table InstanceType -GroupBy CurrentGeneration

CurrentGeneration: False

InstanceType
-----
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge

CurrentGeneration: True

InstanceType
-----
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
c5.4xlarge
c5.9xlarge
...
```

Windows

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi
  Name=processor-info.supported-architecture,Values=x86_64 --query "InstanceTypes[*].
  [InstanceType]" --output text | sort

c5.12xlarge
c5.18xlarge
```

```
c5.24xlarge
c5.2xlarge
c5.4xlarge
c5.9xlarge
c5.large
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object {
    $_.SupportedBootModes -Contains "uefi" -and `
    $_.ProcessorInfo.SupportedArchitectures -eq "x86_64"
  } | `
  Sort-Object InstanceType | `
  Format-Table InstanceType -GroupBy CurrentGeneration

CurrentGeneration: True

InstanceType
-----
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
c5.4xlarge
...
```

Untuk melihat jenis instans yang tersedia yang mendukung Boot Aman UEFI dan mempertahankan variabel non-volatile di Wilayah tertentu

Saat ini, instans bare metal tidak mendukung UEFI Secure Boot dan variabel non-volatile. Gunakan perintah [describe-instance-types](#) seperti yang dijelaskan dalam contoh sebelumnya, tetapi saring instans bare metal dengan menyertakan filter `Name=bare-metal, Values=false`. Untuk informasi tentang UEFI Secure Boot, lihat [UEFI Secure Boot](#).

Gunakan perintah untuk sistem operasi Anda.

Linux

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi
Name=bare-metal,Values=false --query "InstanceTypes[*].[InstanceType]" --output
text | sort
```

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
    Where-Object { `
        $_.SupportedBootModes -Contains "uefi" -and `
        $_.BareMetal -eq $False
    } | `
    Sort-Object InstanceType | `
    Format-Table InstanceType, SupportedBootModes, BareMetal,
    @{Name="SupportedArchitectures";
    Expression={$_.ProcessorInfo.SupportedArchitectures}}
```

InstanceType	SupportedBootModes	BareMetal	SupportedArchitectures
a1.2xlarge	{uefi}	False	arm64
a1.4xlarge	{uefi}	False	arm64
a1.large	{uefi}	False	arm64
a1.medium	{uefi}	False	arm64
a1.xlarge	{uefi}	False	arm64
c5.12xlarge	{legacy-bios, uefi}	False	x86_64
c5.18xlarge	{legacy-bios, uefi}	False	x86_64

Windows

AWS CLI

```
$ aws ec2 describe-instance-types --filters Name=supported-boot-
mode,Values=uefi Name=bare-metal,Values=false Name=processor-info.supported-
```

```
architecture,Values=x86_64 --query "InstanceTypes[*].[InstanceType]" --output text |
sort
```

```
c5.12xlarge
c5.18xlarge
c5.24xlarge
c5.2xlarge
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
  Where-Object { `
    $_.SupportedBootModes -Contains "uefi" -and `
    $_.BareMetal -eq $False -and `
    $_.ProcessorInfo.SupportedArchitectures -eq "x86_64" `
  } | `
  Sort-Object InstanceType | `
  Format-Table InstanceType, SupportedBootModes, BareMetal, `
  @{Name="SupportedArchitectures"; `
  Expression={$_.ProcessorInfo.SupportedArchitectures}}
```

InstanceType	SupportedBootModes	BareMetal	SupportedArchitectures
c5.12xlarge	{legacy-bios, uefi}	False	x86_64
c5.18xlarge	{legacy-bios, uefi}	False	x86_64
c5.24xlarge	{legacy-bios, uefi}	False	x86_64
c5.2xlarge	{legacy-bios, uefi}	False	x86_64
c5.4xlarge	{legacy-bios, uefi}	False	x86_64
c5.9xlarge	{legacy-bios, uefi}	False	x86_64

Persyaratan untuk meluncurkan sebuah instans di UEFI

Untuk meluncurkan instans dalam mode boot UEFI, Anda harus memilih sebuah tipe instans yang mendukung UEFI, dan mengonfigurasi AMI dan sistem operasi untuk UEFI, sebagai berikut:

Jenis instans

Saat meluncurkan instans, Anda harus memilih tipe instans yang mendukung UEFI. Untuk informasi selengkapnya, lihat [Menentukan mode boot yang didukung untuk sebuah tipe instans](#).

AMI

Saat meluncurkan instans, Anda harus memilih AMI yang dikonfigurasi untuk UEFI. AMI harus dikonfigurasi 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 [Menentukan mode boot sistem operasi](#).
- Parameter mode boot AMI – Parameter mode boot AMI harus diatur ke `uefi` atau `uefi-preferred`. Untuk informasi selengkapnya, lihat [Tentukan parameter mode boot suatu AMI](#).

Linux — AWS hanya menyediakan AMI Linux yang dikonfigurasi untuk mendukung UEFI untuk jenis instans berbasis Graviton. Untuk menggunakan Linux pada jenis instans UEFI lainnya, Anda harus [mengkonfigurasi AMI](#), [mengimpor AMI](#) melalui [VM Import/Export](#), atau mengimpor AMI melalui [CloudEndure](#)

Windows — AMI Windows berikut mendukung UEFI:

- 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

Tentukan parameter mode boot suatu AMI

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 suatu AMI (konsol)

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 suatu AMI ketika meluncurkan sebuah instans (konsol)

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 suatu AMI (AWS CLI)

Gunakan operasi [describe-images](#) untuk menentukan mode boot AMI.

```
aws ec2 describe-images --region us-east-1 --image-id ami-0abcdef1234567890

{
  "Images": [
    {
      ...
    ],
    "EnaSupport": true,
    "Hypervisor": "xen",
    "ImageOwnerAlias": "amazon",
    "Name": "UEFI_Boot_Mode_Enabled-Windows_Server-2016-English-Full-
Base-2020.09.30",
    "RootDeviceName": "/dev/sda1",
    "RootDeviceType": "ebs",
    "SriovNetSupport": "simple",
    "VirtualizationType": "hvm",
    "BootMode":
"uefi"
  ]
}
```

Dalam output, bidang `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.

PowerShell

Untuk menentukan parameter mode boot AMI (Alat untuk PowerShell)

Gunakan Cmdlet [Get-EC2Image](#) untuk menentukan mode boot AMI.

```
PS C:\> Get-EC2Image -Region us-east-1 -ImageId ami-0abcdef1234567890 | Format-List
Name, BootMode, TpmSupport

Name      : TPM-Windows_Server-2016-English-Full-Base-2023.05.10
BootMode  : uefi
TpmSupport : v2.0
```

Dalam output, bidang `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.

Menentukan mode boot yang didukung untuk sebuah tipe instans

Anda dapat menggunakan AWS CLI atau Tools PowerShell untuk menentukan mode boot yang didukung dari jenis instance.

Untuk menentukan mode boot yang didukung sebuah tipe instans

Anda dapat menggunakan metode berikut untuk menentukan mode boot yang didukung untuk sebuah tipe instans.

AWS CLI

Anda dapat menggunakan perintah [describe-instance-types](#) untuk menentukan mode boot yang didukung suatu tipe instans. Dengan menyertakan parameter `--query`, Anda dapat menyaring output. Dalam contoh ini, output disaring untuk hanya memunculkan mode boot yang didukung.

Contoh berikut menunjukkan bahwa `m5.2xlarge` mendukung mode boot UEFI dan Legacy BIOS.

```
aws ec2 describe-instance-types --region us-east-1 --instance-types m5.2xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

Output yang diharapkan:

```
[
  [
    "legacy-bios",
    "uefi"
  ]
]
```

Contoh berikut menunjukkan bahwa t2.xlarge hanya mendukung Legacy BIOS.

```
aws ec2 describe-instance-types --region us-east-1 --instance-types t2.xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

Keluaran yang diharapkan

```
[
  [
    "legacy-bios"
  ]
]
```

PowerShell

Anda dapat menggunakan [Get-EC2InstanceType](#) (Alat untuk PowerShell) 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 -Region us-east-1 -InstanceType m5.2xlarge | Format-List
InstanceType, SupportedBootModes
```

Output yang diharapkan:

```
InstanceType      : m5.2xlarge
```

```
SupportedBootModes : {legacy-bios, uefi}
```

Contoh berikut menunjukkan bahwa t2.xlarge hanya mendukung Legacy BIOS.

```
Get-EC2InstanceType -Region us-east-1 -InstanceType t2.xlarge | Format-List  
InstanceType, SupportedBootModes
```

Output yang diharapkan:

```
InstanceType      : t2.xlarge  
SupportedBootModes : {legacy-bios}
```

Menentukan mode boot dari sebuah instans

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 sebuah instans (konsol)

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 sebuah instans (AWS CLI)

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

{
  "Reservations": [
    {
      "Groups": [],
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "ImageId": "ami-0e2063e7f6dc3bee8",
          "InstanceId": "i-1234567890abcdef0",
          "InstanceType": "m5.2xlarge",
          ...
        },
        {
          "BootMode": "uefi",
          "CurrentInstanceBootMode": "uefi"
        }
      ],
      "OwnerId": "1234567890",
      "ReservationId": "r-1234567890abcdef0"
    }
  ]
}
```

PowerShell

Untuk menentukan mode boot dari sebuah instance (Alat untuk PowerShell)

Gunakan Cmdlet [Get-EC2Image](#) untuk menentukan mode boot instans. Anda juga dapat menentukan mode boot AMI yang digunakan untuk membuat instans.

[Get-EC2Image](#) (AWS Tools for Windows PowerShell)

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances | Format-List BootMode,  
CurrentInstanceBootMode, InstanceType, ImageId
```

```
BootMode           : uefi  
CurrentInstanceBootMode : uefi  
InstanceType      : c5a.large  
ImageId           : ami-0265446f88eb4021b
```

Dalam output, parameter berikut menggambarkan mode boot:

- **BootMode** – Mode boot AMI yang digunakan untuk membuat instans.
- **CurrentInstanceBootMode** – Mode boot yang digunakan untuk melakukan boot instans saat diluncurkan atau dimulai.

Menentukan mode boot sistem operasi

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 harus terhubung ke instans Anda menggunakan SSH (instance Linux) atau RDP (instance 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"
```

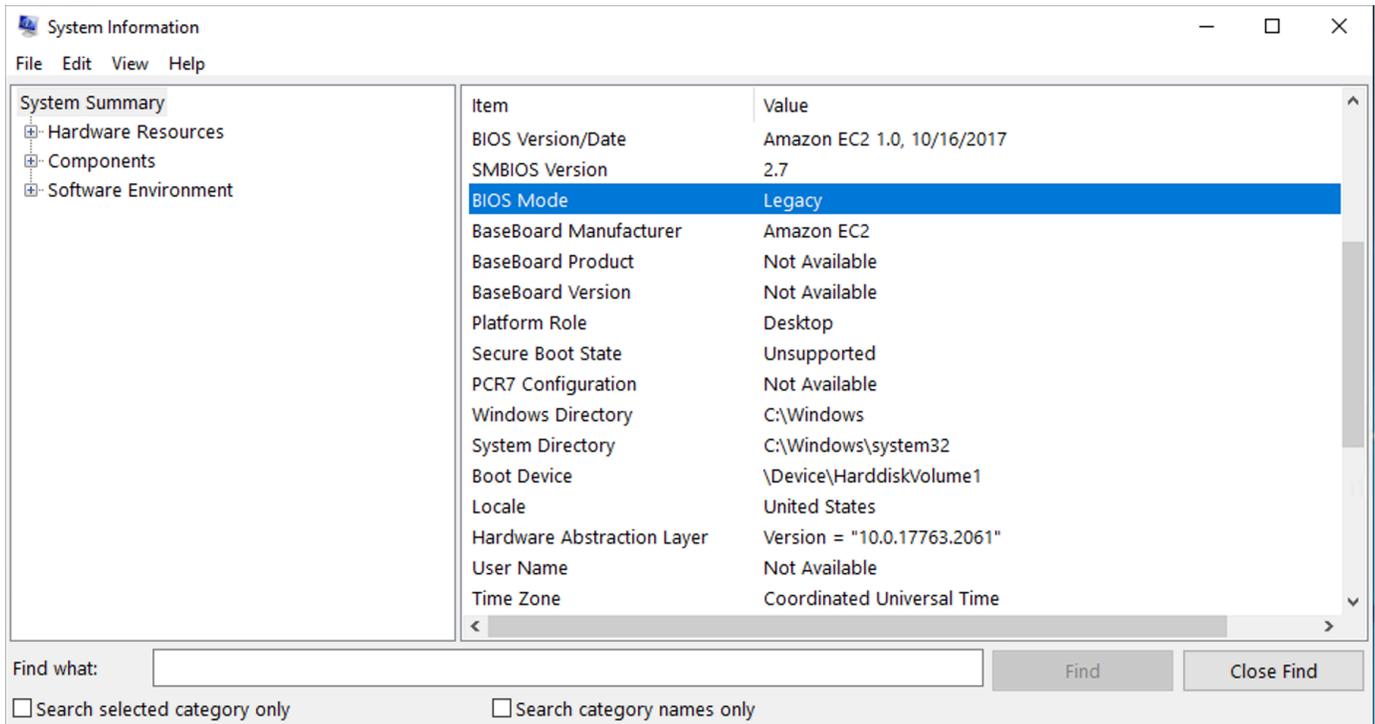
Output yang diharapkan dari sebuah instans yang diboot dalam mode boot UEFI

```
[ 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 sebuah AMI

Saat Anda membuat AMI menggunakan perintah [register-image](#), Anda dapat mengatur mode boot AMI ke `uefi`, `legacy-bios`, atau `uefi-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.

Note

Jika Anda mengatur mode boot AMI ke `uefi-preferred`, sistem operasi harus mendukung kemampuan untuk melakukan boot UEFI dan Legacy BIOS.

Saat ini, Anda tidak dapat menggunakan perintah [register-image](#) untuk membuat AMI yang mendukung [NitroTPM](#) sekaligus UEFI Diutamakan.

⚠ Warning

Beberapa fitur, seperti UEFI Secure Boot, hanya tersedia pada instans yang di-boot di UEFI. Saat Anda menggunakan parameter mode boot AMI `uefi-preferred` dengan tipe instans yang tidak mendukung UEFI, instans akan diluncurkan sebagai Legacy BIOS dan fitur yang bergantung pada UEFI akan dinonaktifkan. Jika Anda mengandalkan ketersediaan fitur yang bergantung pada UEFI, atur parameter mode boot AMI Anda ke `uefi`.

Untuk mengonversi instans berbasis Legacy BIOS ke UEFI, atau instans berbasis UEFI ke Legacy BIOS, Anda perlu melakukan sejumlah langkah: Pertama, modifikasi volume instans dan sistem operasi untuk mendukung mode boot yang dipilih. Kemudian, buat snapshot volume. Terakhir, gunakan [register-image](#) untuk membuat AMI menggunakan snapshot.

Anda tidak dapat mengatur mode boot AMI menggunakan perintah [create-image](#). Dengan [create-image](#), AMI mewarisi mode boot instans EC2 yang digunakan untuk menciptakan AMI tersebut. Sebagai contoh, jika Anda membuat AMI dari instans EC2 yang berjalan pada Legacy BIOS, mode boot AMI akan dikonfigurasi sebagai `legacy-bios`. Jika Anda membuat AMI dari instans EC2 yang diluncurkan menggunakan AMI dengan mode boot yang diatur ke `uefi-preferred`, AMI yang dibuat juga akan memiliki pengaturan mode boot `uefi-preferred`.

⚠ Warning

Menetapkan parameter mode boot AMI tidak secara otomatis mengonfigurasi sistem operasi untuk mode boot tersebut. Sebelum melanjutkan dengan langkah-langkah ini, Anda harus terlebih dahulu membuat perubahan yang sesuai ke volume instans dan sistem operasi untuk mendukung boot menggunakan mode boot yang dipilih; jika tidak, AMI yang dihasilkan tidak akan dapat digunakan. Misalnya, jika Anda mengonversi instance Windows berbasis BioS 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.

Untuk mengatur mode boot sebuah AMI (AWS CLI)

1. Buat perubahan yang sesuai dengan volume instans dan sistem operasi untuk mendukung boot melalui mode boot yang dipilih. Perubahan yang diperlukan adalah perubahan khusus sistem operasi. Untuk informasi lebih lanjut, lihat manual untuk sistem operasi Anda.

Note

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 --region us-east-1 --instance-ids i-1234567890abcdef0
```

Output yang diharapkan

```
...
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/sda1",
        "Ebs": {
          "AttachTime": "",
          "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 --region us-east-1 --volume-id vol-1234567890abcdef0 --
description "add text"
```

Output yang diharapkan

```
{
  "Description": "add text",
  "Encrypted": false,
  "OwnerId": "123",
  "Progress": "",
  "SnapshotId": "snap-01234567890abcdef",
  "StartTime": "",
  "State": "pending",
```

```
"VolumeId": "vol-1234567890abcdef0",
"VolumeSize": 30,
"Tags": []
}
```

4. Perhatikan ID snapshot dalam output dari langkah sebelumnya.
5. Tunggu sampai pembuatan snapshot completed sebelum melanjutkan ke langkah berikutnya. Untuk kueri status snapshot, gunakan perintah [describe-snapshots](#).

```
aws ec2 describe-snapshots --region us-east-1 --snapshot-ids snap-01234567890abcdef
```

Contoh Output

```
{
  "Snapshots": [
    {
      "Description": "This is my snapshot",
      "Encrypted": false,
      "VolumeId": "vol-049df61146c4d7901",
      "State": "completed",
      "VolumeSize": 8,
      "StartTime": "2019-02-28T21:28:32.000Z",
      "Progress": "100%",
      "OwnerId": "012345678910",
      "SnapshotId": "snap-01234567890abcdef",
      ...
    }
  ]
}
```

6. Untuk membuat AMI baru, gunakan perintah [register-image](#). Gunakan ID snapshot yang Anda catat di langkah sebelumnya.
 - Untuk mengatur mode boot ke UEFI, tambahkan parameter `--boot-mode` ke perintah dan tentukan `uefi` sebagai nilainya.

```
aws ec2 register-image \
  --region us-east-1 \
  --description "add description" \
  --name "add name" \
  --block-device-mappings "DeviceName=/dev/
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \
  --architecture x86_64 \
  --root-device-name /dev/sda1 \
  --virtualization-type hvm \
```

```
--ena-support \  
--boot-mode uefi
```

- Untuk mengatur mode boot ke uefi-preferred, tambahkan parameter `--boot-mode` ke perintah dan tentukan `uefi-preferred` sebagai nilainya.

```
aws ec2 register-image \  
  --region us-east-1 \  
  --description "add description" \  
  --name "add name" \  
  --block-device-mappings "DeviceName=/dev/  
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \  
  --architecture x86_64 \  
  --root-device-name /dev/sda1 \  
  --virtualization-type hvm \  
  --ena-support \  
  --boot-mode uefi-preferred
```

Keluaran yang diharapkan

```
{  
  "ImageId": "ami-new_ami_123"  
}
```

7. Untuk memverifikasi bahwa AMI yang baru dibuat memiliki mode boot yang Anda tentukan di langkah sebelumnya, gunakan perintah [describe-images](#).

```
aws ec2 describe-images --region us-east-1 --image-id ami-new_ami_123
```

Keluaran yang diharapkan

```
{  
  "Images": [  
    {  
      "Architecture": "x86_64",  
      "CreationDate": "2021-01-06T14:31:04.000Z",  
      "ImageId": "ami-new_ami_123",  
      "ImageLocation": "",  
      ...  
      "BootMode": "uefi"  
    }  
  ]  
}
```

```
]
}
```

8. Luncurkan instans baru menggunakan AMI yang baru dibuat.

Jika mode boot AMI adalah `uefi` atau `legacy-bios`, instans yang dibuat dari AMI ini akan memiliki mode boot yang sama dengan AMI tersebut. Jika mode boot AMI adalah `uefi-preferred`, instans akan melakukan boot menggunakan UEFI jika tipe instans mendukung UEFI; jika tidak, instans akan boot menggunakan Legacy BIOS. Untuk informasi selengkapnya, lihat [Pertimbangan](#).

9. Untuk memverifikasi bahwa instans yang baru memiliki mode boot yang diharapkan, gunakan perintah [describe-instances](#).

Variabel UEFI

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 berpotensi 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 instans yang diluncurkan pada atau setelah 11 Mei 2022, variabel UEFI yang ditandai sebagai non-volatile akan dipertahankan saat boot ulang dan berhenti/mulai.

- Instans bare metal tidak mempertahankan variabel non-volatile UEFI di seluruh operasi berhenti/memulai instans.

UEFI Secure Boot

UEFI Secure Boot dibangun di atas proses boot aman lama Amazon EC2, dan menyediakan tambahan yang membantu pelanggan mengamankan perangkat lunak dari ancaman defense-in-depth yang bertahan selama reboot. 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.

Topik

- [Cara kerja UEFI Secure Boot](#)
- [Meluncurkan instans dengan dukungan UEFI Secure Boot](#)
- [Verifikasi apakah instans diaktifkan untuk UEFI Secure Boot](#)
- [Membuat AMI Linux untuk mendukung UEFI Secure Boot](#)
- [Bagaimana gumpalan AWS biner dibuat](#)

Cara kerja UEFI Secure Boot

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 mengelola dbx yang tersedia untuk umum untuk banyak biner dan sertifikat yang diketahui buruk 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 keluar. SetupMode

Meluncurkan instans dengan dukungan UEFI Secure Boot

Saat Anda [meluncurkan instans](#) dengan prasyarat berikut, instans akan secara otomatis memvalidasi biner boot UEFI terhadap basis data UEFI Secure Boot-nya. 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. Biasanya, UEFI Secure Boot dikonfigurasi sebagai bagian dari AMI. Jika Anda membuat AMI baru dengan parameter yang berbeda dari AMI dasar, seperti mengubah `UefiData` dalam AMI, Anda dapat menonaktifkan UEFI Secure Boot.

Prasyarat

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 [Membuat AMI Linux untuk mendukung UEFI Secure Boot](#).

AMI Windows

Untuk meluncurkan instance Windows, AMI Windows harus mengaktifkan UEFI Secure Boot.

AMI Windows berikut ini telah dikonfigurasi sebelumnya untuk mengaktifkan UEFI Secure Boot dengan kunci Microsoft:

- TPM-Windows_Server-2022-English-Core-Base
- TPM-Windows_Server-2022-English-Full-Base

- TPM-Windows_Server-2022-English-Full-SQL_2022_Enterprise
- TPM-Windows_Server-2022-English-Full-SQL_2022_Standard
- TPM-Windows_Server-2019-English-Core-Base
- TPM-Windows_Server-2019-English-Full-Base
- TPM-Windows_Server-2019-Inggris-penuh-SQL_2019_Enterprise
- TPM-Windows_Server-2019-English-Full-SQL_2019_Standard
- TPM-Windows_Server-2016-English-Core-Base
- TPM-Windows_Server-2016-English-Full-Base

Saat ini, kami tidak mendukung mengimpor Windows dengan UEFI Secure Boot menggunakan perintah [import-image](#).

Jenis instans

- Didukung: Semua tipe instans virtual yang mendukung UEFI juga mendukung UEFI Secure Boot. Untuk tipe instans yang mendukung UEFI Secure Boot, lihat [Pertimbangan](#).
- Tidak didukung: Tipe instans bare metal tidak mendukung UEFI Secure Boot.

Verifikasi apakah instans 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 Amazon Linux 2, lihat <https://docs.aws.amazon.com/linux/al2/ug/find-install-software.html>. Untuk distribusi Linux lainnya, lihat dokumentasi spesifiknya.

Untuk memverifikasi apakah sebuah instans Linux diaktifkan untuk UEFI Secure Boot

Jalankan perintah berikut sebagai `root` pada instans.

```
mokutil --sb-state
```

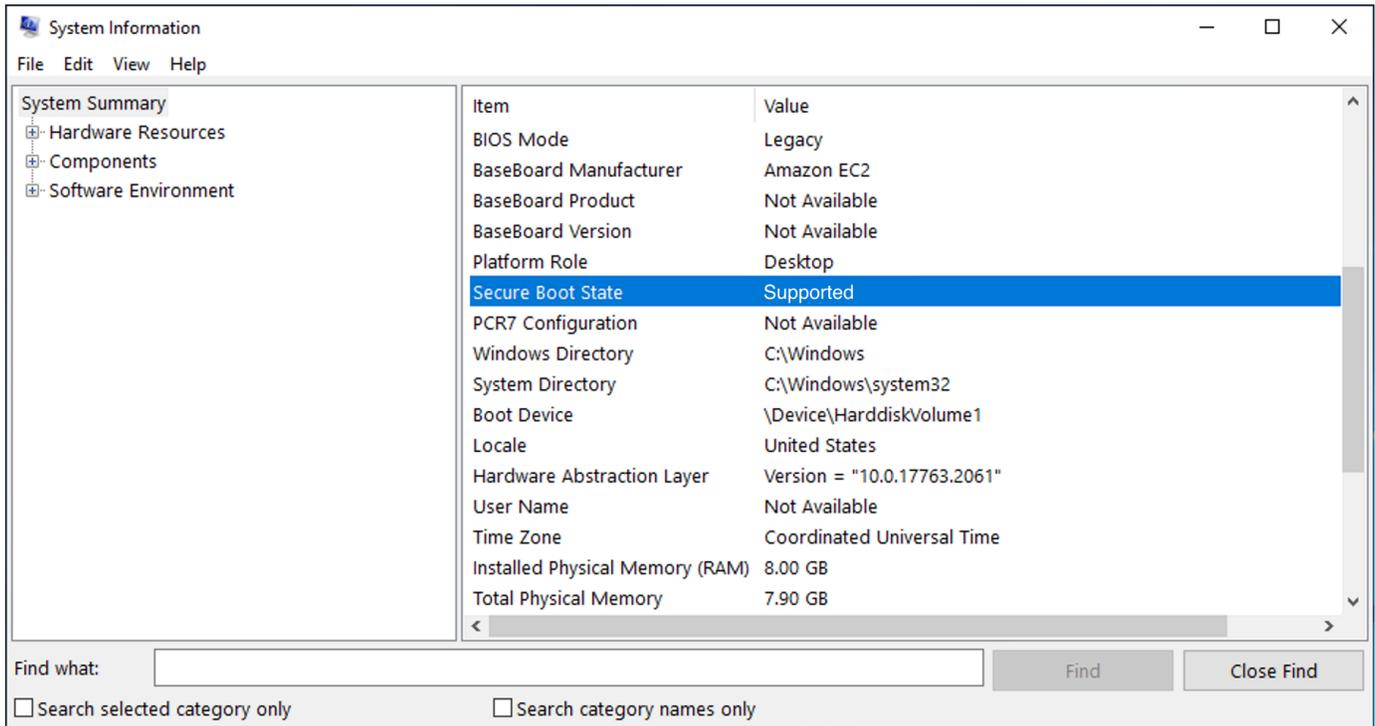
Output yang diharapkan:

- 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. Buka alat msinfo32.
2. Periksa bidang Kondisi Secure Boot. Didukung menunjukkan bahwa UEFI Secure Boot diaktifkan.



Anda juga dapat menggunakan Windows PowerShell Cmdlet `Confirm-SecureBootUEFI` untuk memeriksa status Boot Aman. Untuk informasi selengkapnya tentang cmdlet, lihat [Konfirmasi-SecureBoot UEFI](#) di situs web Microsoft Documentation.

Membuat AMI Linux untuk mendukung UEFI Secure Boot

Prosedur berikut ini menjelaskan cara membuat penyimpanan variabel UEFI Anda sendiri untuk boot aman dengan kunci privat kustom. Amazon Linux mendukung UEFI Secure Boot dimulai dengan AL2023 versi 2023.1. Untuk informasi selengkapnya, lihat [UEFI Secure Boot](#) di Panduan Pengguna AL2023.

⚠ Important

Prosedur membuat AMI untuk mendukung UEFI Secure Boot berikut ini ditujukan hanya untuk pengguna tingkat lanjut. 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](#) AWS CLI
- 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 untuk mendukung UEFI Secure Boot, pertama buat tiga pasang kunci, lalu selesaikan Opsi A atau Opsi B:

- [Buat tiga pasang kunci](#)
- [Opsi A: Tambahkan kunci ke penyimpanan variabel dari dalam instans](#)
- [Opsi B: Buat gumpalan biner yang berisi penyimpanan variabel yang telah diisi sebelumnya](#)

Note

Instruksi ini hanya dapat digunakan untuk membuat AMI Linux. Jika Anda memerlukan AMI Windows, gunakan salah satu AMI Windows yang didukung. Untuk informasi selengkapnya, lihat [Meluncurkan instans dengan dukungan UEFI Secure Boot](#).

Buat tiga pasang 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.

Topik

- [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.

Standar X.509 digunakan untuk membuat pasangan kunci. Untuk informasi tentang standar yang digunakan, 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 dibandingkan *Kunci platform*.

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 untuk 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 ArchWiki Pemeliharaan dan/atau kontributor. ArchWiki

Opsi A: Tambahkan kunci ke penyimpanan variabel dari dalam instans

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.

Langkah-langkah Opsi A:

- [Langkah 1: Luncurkan instans yang akan mendukung UEFI Secure Boot](#)

- [Langkah 2: Konfigurasi 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 suatu AMI](#).

Perhatikan bahwa AWS hanya menyediakan AMI Linux yang dikonfigurasi untuk mendukung UEFI untuk jenis instans berbasis Graviton. AWS saat ini tidak menyediakan AMI Linux x86_64 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 sebuah AMI](#).

- 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 [Pertimbangan](#).
- 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-0123456789example
```

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: Konfigurasi 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 [Verifikasi apakah instans diaktifkan untuk UEFI Secure Boot](#).

Anda sekarang dapat mengekspor penyimpanan variabel UEFI Anda dengan perintah CLI [get-instance-uefi-data](#), atau Anda melanjutkan ke langkah berikutnya dan menandatangani gambar boot Anda untuk boot ulang ke instans dengan UEFI Secure Boot diaktifkan.

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 AMI yang didukung Amazon EBS-Backed](#). Untuk petunjuk 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.

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.

 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/aws-labs/python-uefivars
```

2. Buat penyimpanan variabel baru dari kunci Anda dengan menjalankan perintah berikut. Ini akan membuat gumpalan biner base64 di bin `your_binary_blob`. Alat ini juga mendukung pembaruan gumpalan biner melalui parameter `-I`.

```
./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
```

Bagaimana gumpalan AWS biner dibuat

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 gumpalan AWS 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. Unduh permintaan perubahan dbx yang diperbarui dari tautan berikut.

```
https://uefi.org/revocationlistfile
```

7. Permintaan perubahan dbx yang Anda unduh pada langkah sebelumnya sudah ditandatangani dengan Microsoft KEK, jadi Anda perlu menghapus atau membongkarnya. Anda dapat menggunakan tautan berikut.

```
https://gist.github.com/out0xb2/f8e0bae94214889a89ac67fceb37f8c0
```

```
https://support.microsoft.com/en-us/topic/microsoft-guidance-for-applying-secure-boot-dbx-update-e3b9e4cb-a330-b3ba-a602-15083965d9ca
```

8. Buat penyimpanan variabel UEFI menggunakan skrip `uefivars.py`.

```
./uefivars.py -i none -o aws -0 uefiblob-microsoft-keys-empty-pk.bin -P ~/PK.esl -K ~/MS_Win_KEK.esl --db ~/MS_db.esl --dbx ~/dbx-2021-April.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 -0 uefiblob-microsoft-keys-empty-pk.bin -P ~/PK.esl -K ~/MS_Win_KEK.esl --db ~/MS_db.esl --dbx ~/dbx-2021-April.bin
```

Keluaran yang diharapkan

```
Replacing PK
Replacing KEK
Replacing db
Replacing dbx
```

Temukan AMI

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

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

- Wilayah — ID AMI unik untuk setiap AWS Wilayah.
- Sistem operasi

- Arsitektur: 32-bit (i386), 64-bit (x86_64), atau 64-bit ARM (arm64)
- Tipe perangkat root: Amazon EBS atau penyimpanan instans
- Penyedia (misalnya, Amazon Web Services)
- Perangkat lunak tambahan (misalnya, SQL Server)

Ada berbagai cara untuk menemukan AMI yang memenuhi kebutuhan Anda. Topik ini menjelaskan cara menemukan AMI menggunakan konsol Amazon EC2, AWS CLI, AWS Tools for Windows PowerShell, dan AWS Systems Manager.

Topik

- [Temukan AMI menggunakan konsol Amazon EC2](#)
- [Temukan AMI menggunakan AWS CLI](#)
- [Temukan AMI menggunakan AWS Tools for Windows PowerShell](#)
- [Temukan AMI menggunakan parameter Systems Manager](#)
- [Temukan AMI terbaru menggunakan Systems Manager](#)
- [Informasi lebih lanjut untuk menemukan AMI](#)

Temukan AMI menggunakan konsol Amazon EC2

Anda dapat menemukan AMI menggunakan konsol Amazon EC2. Anda dapat memilih dari daftar AMI saat Anda menggunakan wizard peluncuran instans untuk meluncurkan suatu instans, atau Anda dapat mencari dari semua AMI yang tersedia menggunakan halaman Gambar.

Untuk menemukan AMI 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 AWS Wilayah.
3. Dari dasbor konsol, pilih Luncurkan instans.
4. (Konsol baru) Di bawah Gambar Aplikasi dan OS (Amazon Machine Image), pilih Mulai Cepat, pilih sistem operasi (OS) untuk instans Anda, lalu dari Amazon Machine Image (AMI), pilih dari salah satu AMI yang umum digunakan dalam daftar tersebut. 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 \(Gambar Mesin Amazon\)](#).

(Konsol lama) Pada tab Mulai Cepat, pilih dari salah satu AMI yang biasanya digunakan dalam daftar. Jika Anda tidak melihat AMI yang ingin Anda gunakan, pilih tab AMI Saya, AWS Marketplace, atau AMI Komunitas untuk mencari AMI tambahan. Untuk informasi selengkapnya, lihat [Langkah 1: Pilih Amazon Machine Image \(AMI\)](#).

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 AWS Wilayah.
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 tipe perangkat 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 [Penyimpanan untuk perangkat root](#).
7. Untuk meluncurkan instans dari AMI ini, pilih dan pilih Peluncuran instans dari gambar. Untuk informasi selengkapnya tentang meluncurkan instans menggunakan konsol, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#). Jika Anda belum siap untuk meluncurkan instans, catatlah ID AMI untuk nanti.

Temukan AMI menggunakan AWS CLI

Anda dapat menggunakan [AWS CLI perintah deskripsi-gambar](#) untuk mencantumkan hanya AMI yang sesuai dengan kebutuhan Anda. Setelah menemukan AMI yang sesuai dengan kebutuhan Anda, catat ID-nya sehingga Anda dapat menggunakannya untuk meluncurkan instans. Untuk

informasi selengkapnya, lihat [Meluncurkan instans Anda](#) di Panduan Pengguna AWS Command Line Interface .

Perintah [describe-images](#) mendukung parameter penyaringan. Misalnya, gunakan parameter `--owners` untuk menampilkan AMI publik yang dimiliki Amazon.

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

Anda dapat menambahkan filter berikut ke perintah sebelumnya untuk hanya menampilkan AMI Windows.

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

Anda dapat menambahkan filter berikut ke perintah sebelumnya untuk hanya menampilkan AMI yang didukung Amazon EBS.

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

Important

Jika Anda menghilangkan `--owners` parameter dari `describe-images` perintah, semua gambar dikembalikan yang Anda miliki izin peluncurannya, terlepas dari kepemilikannya.

Temukan AMI menggunakan AWS Tools for Windows PowerShell

Anda dapat menggunakan PowerShell cmdlet untuk mencantumkan hanya AMI Windows yang sesuai dengan kebutuhan Anda. Untuk informasi dan contoh, lihat [Menemukan Gambar Mesin Amazon Menggunakan Windows PowerShell](#) di Panduan AWS Tools for Windows PowerShell Pengguna.

Setelah menemukan AMI yang sesuai dengan kebutuhan Anda, catat ID-nya sehingga Anda dapat menggunakannya untuk meluncurkan instans. Untuk informasi selengkapnya, lihat [Meluncurkan Instans Amazon EC2 Menggunakan Windows PowerShell](#) di AWS Tools for Windows PowerShell Panduan Pengguna.

Temukan AMI menggunakan parameter Systems Manager

Saat meluncurkan instans menggunakan wizard instans peluncuran EC2 di konsol Amazon EC2, Anda dapat memilih AMI dari daftar (dijelaskan [Temukan AMI menggunakan konsol Amazon EC2](#) dalam), atau Anda dapat memilih parameter AWS Systems Manager 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 .

Topik

- [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 Systems Manager yang mengarah ke ID AMI di wizard instans peluncuran, Anda harus menambahkan izin berikut ke kebijakan IAM Anda:

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

Anda juga dapat membatasi akses ke parameter Systems Manager tertentu. Untuk informasi selengkapnya dan contoh kebijakan IAM, lihat [Contoh: 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.

Meluncurkan instans menggunakan parameter Systems Manager

Anda dapat meluncurkan instans menggunakan konsol atau AWS CLI. Alih-alih menentukan ID AMI, Anda dapat menentukan AWS Systems Manager parameter yang menunjuk ke ID AMI.

New console

Untuk menemukan AMI menggunakan parameter Systems Manager (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. 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Old console

Untuk menemukan AMI menggunakan parameter Systems Manager (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. Dari dasbor konsol, pilih Luncurkan instans.
4. Pilih Pencarian berdasarkan parameter System Manager (di kanan atas).
5. Untuk Parameter System Manager, pilih parameter. ID AMI terkait muncul di samping Saat ini menyelesaikan.
6. Pilih Cari. AMI yang cocok dengan ID AMI muncul dalam daftar.
7. Pilih AMI dari daftar, lalu pilih Pilih.

Untuk informasi selengkapnya tentang peluncuran instans dari AMI menggunakan wizard peluncuran instans, lihat [Langkah 1: Pilih Amazon Machine Image \(AMI\)](#).

Untuk meluncurkan instance menggunakan AWS Systems Manager parameter, bukan ID AMI (AWS CLI)

Contoh berikut ini menggunakan parameter System Manager `golden-ami` untuk meluncurkan instans `m5.xlarge`. Parameter menunjuk ke ID AMI.

Untuk menetapkan parameter dalam perintah, gunakan sintaksis berikut:

`resolve:ssm:/parameter-name`, di mana `resolve:ssm` adalah awalan standar dan `parameter-name` adalah nama parameter unik. Perhatikan bahwa nama parameter bersifat 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.

Dalam contoh ini, parameter `--count` dan `--security-group` tidak disertakan. Untuk `--count`, default-nya adalah 1. Jika Anda memiliki VPC default dan grup keamanan default, keduanya akan digunakan.

```
aws ec2 run-instances
  --image-id resolve:ssm:/golden-ami
  --instance-type m5.xlarge
  ...
```

Untuk meluncurkan instance menggunakan versi tertentu dari AWS Systems Manager parameter (AWS CLI)

Parameter Systems Manager memiliki dukungan versi. Setiap iterasi parameter diberi nomor versi unik. Anda dapat merujuk ke versi parameter sebagai berikut `resolve:ssm:parameter-`

`name:version`, di mana `version` adalah nomor versi unik. Secara default, versi terbaru parameter digunakan ketika tidak ada versi yang ditentukan.

Contoh berikut ini menggunakan parameter versi 2.

Dalam contoh ini, parameter `--count` dan `--security-group` tidak disertakan. Untuk `--count`, default-nya adalah 1 jika Anda memiliki VPC default dan grup keamanan default, keduanya akan digunakan.

```
aws ec2 run-instances
  --image-id resolve:ssm:/golden-ami:2
  --instance-type m5.xlarge
  ...
```

Untuk meluncurkan instance menggunakan parameter publik yang disediakan oleh AWS

Systems Manager menyediakan parameter publik untuk AMI publik yang disediakan oleh AWS. Anda dapat menggunakan parameter publik saat meluncurkan instance untuk memastikan bahwa Anda menggunakan AMI terbaru.

Untuk informasi selengkapnya, lihat [Temukan AMI terbaru menggunakan Systems Manager](#).

Temukan AMI terbaru menggunakan Systems Manager

AWS Systems Manager menyediakan parameter publik untuk AMI publik yang dikelola oleh AWS. Anda dapat menggunakan parameter publik saat meluncurkan instance untuk memastikan bahwa Anda menggunakan AMI terbaru. Misalnya, parameter publik `/aws/service/ami-amazon-linux-latest/a12023-ami-kernel-default-arm64` tersedia di semua Wilayah dan selalu menunjuk ke versi terbaru 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 melihat daftar semua AMI Linux atau Windows di AWS Wilayah saat ini

Gunakan AWS CLI perintah [get-parameters-by-path](#) berikut untuk melihat daftar semua AMI Linux atau Windows di Wilayah saat ini. AWS Nilai untuk `--path` parameter berbeda untuk Linux dan Windows.

Untuk Linux:

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

Untuk Windows:

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

Untuk meluncurkan suatu instans menggunakan parameter publik

Contoh berikut menentukan parameter publik Systems Manager untuk ID gambar untuk meluncurkan instance menggunakan AMI Amazon Linux 2023 terbaru.

Untuk menetapkan parameter dalam perintah, gunakan sintaksis berikut: `resolve:ssm:public-parameter`, di mana `resolve:ssm` adalah awalan standar dan `public-parameter` adalah jalan dan nama parameter publik.

Dalam contoh ini, parameter `--count` dan `--security-group` tidak disertakan. Untuk `--count`, default-nya adalah 1. Jika Anda memiliki VPC default dan grup keamanan default, keduanya akan digunakan.

```
aws ec2 run-instances \  
  --image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-  
  default-x86_64 \  
  --instance-type m5.xlarge \  
  --key-name MyKeyPair
```

Untuk informasi selengkapnya, lihat [Bekerja dengan parameter publik](#) di Panduan Pengguna AWS Systems Manager .

Untuk contoh yang menggunakan parameter Systems Manager, lihat [Kueri untuk ID AMI Amazon Linux terbaru Menggunakan AWS Systems Manager Parameter Store](#) dan [Query untuk AMI Windows Terbaru Menggunakan AWS Systems Manager Parameter Store](#).

Informasi lebih lanjut untuk menemukan AMI

Untuk menemukan AMI Amazon Linux 2023, lihat [AL2023 di Amazon EC2 di Panduan Pengguna Amazon Linux 2023](#).

Untuk menemukan AMI Ubuntu, lihat [Amazon EC2 AMI Locator di situs web Canonical Ubuntu](#).

Untuk menemukan RHEL AMI, lihat [Red Hat Enterprise Linux Images \(AMI\) Tersedia di Amazon Web Services \(AWS\)](#) di situs web Red Hat.

AMI bersama

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 deskripsi-gambar](#) untuk mengidentifikasi AMI publik yang berasal dari penyedia terverifikasi. Gambar publik yang dimiliki oleh Amazon atau mitra terverifikasi memiliki pemilik alias, yaitu `amazon` 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 [Memulai sebagai penjual](#) dan [produk berbasis AMI](#) di Panduan Penjual AWS Marketplace .

Topik AMI bersama

- [Mencari AMI bersama](#)
- [Menjadikan AMI publik](#)
- [Membagikan AMI dengan organisasi atau unit organisasi tertentu](#)
- [Membagikan AMI kepada akun AWS tertentu](#)
- [Batalkan AMI yang dibagikan dengan Anda Akun AWS](#)
- [Menggunakan bookmark](#)
- [Panduan untuk AMI Linux bersama](#)

Jika Anda mencari informasi tentang topik lain

- Untuk informasi tentang membuat AMI, lihat [the section called “Buat AMI Linux yang didukung penyimpanan instans”](#) atau [the section called “Buat AMI yang didukung Amazon EBS-Backed”](#).
- Untuk informasi tentang membangun, mengirim, dan memelihara aplikasi Anda di AWS Marketplace, lihat [Dokumentasi AWS Marketplace](#).

Mencari AMI bersama

Anda dapat menggunakan konsol Amazon EC2 atau baris perintah untuk mencari AMI bersama.

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 [Menyalin AMI](#).

Tugas

- [Mencari AMI bersama \(konsol\)](#)
- [Mencari AMI bersama \(AWS CLI\)](#)
- [Temukan AMI bersama \(Alat untuk Windows PowerShell\)](#)
- [Menggunakan AMI bersama](#)

Mencari AMI bersama (konsol)

Untuk mencari AMI privat bersama menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di panel navigasi, pilih AMI.
3. Di filter pertama, pilih Gambar privat. Semua AMI yang telah dibagikan kepada Anda telah tercantum. Untuk menyusun secara terperinci pencarian Anda, pilih bilah Pencarian dan gunakan opsi filter yang tersedia pada menu.

Untuk mencari AMI privat bersama menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih AMI.
3. Di filter pertama, pilih Gambar publik. Untuk menyusun secara terperinci pencarian Anda, pilih bidang Pencarian dan gunakan opsi filter yang tersedia pada menu.

Untuk mencari AMI publik bersama Amazon menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih AMI.
3. Di filter pertama, pilih Gambar publik.
4. Pilih bidang Pencarian, lalu dari opsi menu yang muncul, pilih Alias pemilik, lalu =, lalu amazon untuk hanya menampilkan gambar publik Amazon.

Untuk mencari AMI publik bersama dari penyedia terverifikasi menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Katalog AMI.
3. Pilih AMI Komunitas.
4. Label Penyedia terverifikasi menunjukkan AMI yang berasal dari Amazon atau mitra terverifikasi.

Mencari AMI bersama (AWS CLI)

Gunakan perintah [describe-images](#) (AWS CLI) untuk mencantumkan AMI. Anda dapat mempersempit daftar tipe AMI yang menarik bagi Anda, seperti yang ditunjukkan dalam contoh berikut.

Contoh: Cantumkan semua AMI publik

Perintah berikut mencantumkan semua AMI publik, termasuk AMI publik yang Anda miliki.

```
aws ec2 describe-images --executable-users all
```

Contoh: Cantumkan 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
```

Contoh: Cantumkan AMI yang dimiliki penyedia terverifikasi

Perintah berikut ini mencantumkan AMI yang dimiliki oleh penyedia terverifikasi. AMI publik yang dimiliki oleh penyedia terverifikasi (baik Amazon atau mitra terverifikasi) memiliki pemilik beralias, yang muncul sebagai amazon 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
```

Contoh: Cantumkan AMI yang dimiliki oleh sebuah akun

Perintah berikut ini mencantumkan AMI yang dimiliki oleh Akun AWS tertentu.

```
aws ec2 describe-images --owners 123456789012
```

Contoh: Persempit 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 AMI yang didukung EBS.

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

Temukan AMI bersama (Alat untuk Windows PowerShell)

Gunakan [Get-EC2Image](#) perintah (Alat untuk Windows PowerShell) untuk mencantumkan AMI. Anda dapat mempersempit daftar tipe AMI yang menarik bagi Anda, seperti yang ditunjukkan dalam contoh berikut.

Contoh: Cantumkan semua AMI publik

Perintah berikut mencantumkan semua AMI publik, termasuk AMI publik yang Anda miliki.

```
PS C:\> Get-EC2Image -ExecutableUser all
```

Contoh: Cantumkan 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.

```
PS C:\> Get-EC2Image -ExecutableUser self
```

Contoh: Cantumkan AMI yang dimiliki penyedia terverifikasi

Perintah berikut ini mencantumkan AMI yang dimiliki oleh penyedia terverifikasi. AMI publik yang dimiliki oleh penyedia terverifikasi (baik Amazon atau mitra terverifikasi) memiliki pemilik beralias, yang muncul sebagai `amazon` 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.

```
PS C:\> Get-EC2Image -Owner amazon aws-marketplace
```

Contoh: Cantumkan AMI yang dimiliki oleh sebuah akun

Perintah berikut ini mencantumkan AMI yang dimiliki oleh Akun AWS tertentu.

```
PS C:\> Get-EC2Image -Owner 123456789012
```

Contoh: Persempit 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 AMI yang didukung EBS.

```
-Filter @{ Name="root-device-type"; Values="ebs" }
```

Menggunakan AMI bersama

Sebelum Anda menggunakan AMI bersama, lakukan langkah-langkah berikut untuk mengonfirmasi bahwa tidak ada kredensial bawaan yang akan memungkinkan akses yang tidak diinginkan ke instans Anda oleh pihak ketiga dan tidak ada pembuatan log jarak jauh terkonfigurasi yang dapat memindahkan 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](#).

Menjadikan AMI publik

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 mengizinkan hanya akun tertentu yang menggunakan AMI Anda untuk meluncurkan instans, lihat [Membagikan AMI kepada akun AWS tertentu](#).

Daftar Isi

- [Pertimbangan](#)
- [Bagikan AMI dengan semua AWS akun \(bagikan secara publik\)](#)
- [Memblokir akses publik ke AMI Anda](#)

Pertimbangan

Pertimbangkan hal berikut sebelum menjadikan AMI publik.

- Kepemilikan — Untuk membuat AMI 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 [Menyalin AMI](#).
- 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 [Panduan untuk 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 untuk memindahkan penghentian ke tanggal berikutnya, Anda harus menjadikan AMI pribadi dengan hanya membagikannya dengan spesifik. Akun AWS](#)
- Hapus AMI usang — Setelah AMI publik mencapai tanggal penghentian, jika tidak ada instance baru yang diluncurkan dari AMI selama enam bulan atau lebih, AWS akhirnya menghapus properti berbagi publik sehingga AMI usang tidak muncul di daftar AMI publik.
- Penagihan — Anda tidak ditagih ketika 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 pemilik, diizinkan untuk menggunakan AMI itu untuk meluncurkan instance. Dengan memodifikasi `launchPermission` properti AMI, Anda dapat membuat AMI menjadi publik (yang memberikan izin peluncuran ke 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 pemilik, diizinkan untuk menggunakan AMI itu untuk meluncurkan instance. Dengan memodifikasi `launchPermission` properti AMI, Anda dapat membuat AMI menjadi publik (yang memberikan izin peluncuran ke 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.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
  launchPermission -OperationType add -UserGroup all
```

2. Untuk memverifikasi izin peluncuran AMI, gunakan perintah [Get-EC2ImageAttribute](#) berikut.

```
PS C:\> 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.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
  launchPermission -OperationType remove -UserGroup all
```

Memblokir akses publik ke AMI Anda

Untuk mencegah AMI Anda dibagikan secara publik, Anda dapat mengaktifkan blokir akses publik untuk AMI. Pengaturan ini diaktifkan di tingkat akun, tetapi Anda harus mengaktifkannya Wilayah AWS di masing-masing tempat Anda ingin mencegah pembagian publik AMI Anda.

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 membagikan 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.

Anda dapat membatasi izin IAM untuk pengguna administrator sehingga hanya mereka yang dapat mengaktifkan atau menonaktifkan blokir akses publik untuk AMI.

Daftar Isi

- [Pengaturan default](#)
- [Izin IAM yang diperlukan](#)
- [Mengaktifkan blokir akses publik untuk AMI](#)
- [Menonaktifkan blokir akses publik untuk AMI](#)
- [Melihat status blokir akses publik 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
Akun yang ada tanpa AMI publik ¹	Aktif
	Nonaktif

AWS akun

Blokir akses publik untuk pengaturan default AMI

Akun yang ada dengan satu atau lebih AMI publik

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.

Izin IAM yang diperlukan

Untuk menggunakan blokir akses publik untuk AMI, Anda harus memiliki izin IAM berikut:

- `EnableImageBlockPublicAccess`
- `DisableImageBlockPublicAccess`
- `GetImageBlockPublicAccessState`

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.

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. Jika dasbor tidak ditampilkan, di panel navigasi, pilih Dasbor EC2.
4. Di bawah Atribut akun, 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 Perbarui.

Note

API dapat memakan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, nilainya akan Berbagi publik baru diizinkan. Ketika API telah menyelesaikan konfigurasi, nilai akan secara otomatis berubah menjadi Berbagi publik baru diblokir.

AWS CLI

Untuk mengaktifkan blokir akses publik untuk AMI di Wilayah tertentu

Gunakan perintah [enable-image-block-public-access](#) dan tentukan Wilayah tempat blokir akses publik untuk AMI ingin diaktifkan. Untuk parameter `--image-block-public-access-state`, tentukan `block-new-sharing`.

```
aws ec2 enable-image-block-public-access \
  --region us-east-1 \
  --image-block-public-access-state block-new-sharing
```

Output yang diharapkan

```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Note

API dapat memakan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, jika Anda menjalankan perintah [get-image-block-public-access-state](#), responsnya adalah `unblocked`. Ketika API telah menyelesaikan konfigurasi, responsnya akan menjadi `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 blokir akses publik untuk AMI Wilayah AWS di masing-masing tempat Anda ingin mengizinkan berbagi AMI Anda secara publik.

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. Jika dasbor tidak ditampilkan, di panel navigasi, pilih Dasbor EC2.
4. Di bawah Atribut akun, 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 Perbarui.
7. Masukkan **confirm** saat diminta konfirmasi, lalu pilih Izinkan berbagi publik.

Note

API dapat memakan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, nilainya akan Berbagi publik baru diblokir. Ketika API telah menyelesaikan konfigurasi, nilai akan secara otomatis berubah menjadi Berbagi publik baru diizinkan.

AWS CLI

Untuk menonaktifkan blokir akses publik untuk AMI di Wilayah tertentu

Gunakan perintah [disable-image-block-public-access](#) dan tentukan Wilayah tempat blokir akses publik untuk AMI ingin dinonaktifkan.

```
aws ec2 disable-image-block-public-access --region us-east-1
```

Output yang diharapkan

```
{  
  "ImageBlockPublicAccessState": "unblocked"  
}
```

Note

API dapat memakan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, jika Anda menjalankan perintah [get-image-block-public-access-state](#), responsnya adalah `block-new-sharing`. Ketika API telah menyelesaikan konfigurasi, responsnya akan menjadi `unblocked`.

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.

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. Jika dasbor tidak ditampilkan, di panel navigasi, pilih Dasbor EC2.
4. Di bawah Atribut akun, 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 blokir akses publik untuk AMI di Wilayah tertentu

Gunakan perintah [get-image-block-public-access](#) dan tentukan Wilayah untuk mendapatkan status blokir akses publik untuk AMI.

```
aws ec2 get-image-block-public-access-state --region us-east-1
```

Output yang diharapkan – Nilainya adalah `block-new-sharing` atau `unblocked`.

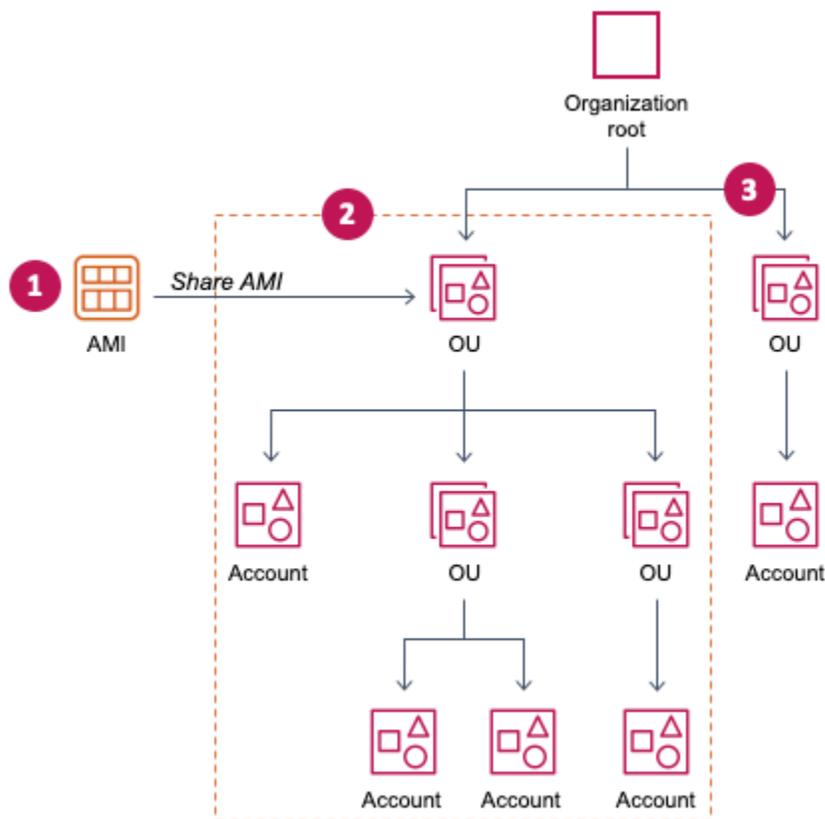
```
{
  "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Membagikan AMI dengan organisasi atau unit organisasi tertentu

[AWS Organizations](#) adalah 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.



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 membagikan AMI, tag yang ditentukan pengguna tidak tersedia untuk organisasi atau OU mana pun Akun AWS yang dengannya 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::account-id:organization/organization-id`
- ARN OU: `arn:aws:organizations::account-id:ou/organization-id/ou-id`

Di mana:

- *account-id* adalah nomor akun manajemen 12 digit, misalnya, 123456789012. 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 [Mendapatkan ARN](#).
- *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 [Amazon Resource Names \(ARN\)](#) 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 terkelola default AWS .
 - 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 [Menyalin AMI](#).
- 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 ketika AMI Anda digunakan oleh orang lain Akun AWS untuk meluncurkan instans. Akun yang meluncurkan instans menggunakan AMI akan dikenai biaya untuk instans yang diluncurkan.

Mengizinkan organisasi dan OU untuk menggunakan kunci KMS

Jika Anda berbagi AMI yang didukung oleh snapshot terenkripsi, Anda juga harus mengizinkan organisasi atau OU untuk menggunakan AWS KMS keys yang digunakan untuk mengenkripsi snapshot.

Gunakan `aws:PrincipalOrgPaths` tombol `aws:PrincipalOrgID` dan untuk membandingkan AWS Organizations jalur untuk kepala sekolah yang membuat permintaan ke jalur dalam kebijakan. Prinsipal itu dapat berupa pengguna, peran IAM, pengguna federasi, atau pengguna Akun AWS root. Dalam kebijakan, kunci ketentuan ini memastikan bahwa pemohon adalah anggota akun dalam root organisasi tertentu atau OU di AWS Organizations. Untuk contoh pernyataan kondisi lainnya, lihat [aws:PrincipalOrgID](#) dan [aws:PrincipalOrgPaths](#) di Panduan Pengguna IAM.

Untuk informasi tentang mengedit kebijakan kunci, lihat [Mengizinkan pengguna di akun lain menggunakan kunci KMS](#) di Panduan AWS Key Management Service Pengembang.

Untuk memberikan izin kepada organisasi atau OU 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*"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:PrincipalOrgID": "o-123example"
    }
  }
}
```

Untuk berbagi kunci KMS dengan beberapa OU, 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*"
  ],
  "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/*"
      ]
    }
  }
}

```

Membagikan AMI

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk berbagi AMI dengan organisasi atau OU.

Membagikan AMI (konsol)

Untuk berbagi AMI dengan organisasi atau OU menggunakan konsol

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 Organisasi/OU berbagi, pilih Tambah ARN Organisasi/OU.

6. Untuk ARN Organisasi/OU, masukkan ARN organisasi atau ARN OU di mana Anda ingin berbagi 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.

 Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, dan sistem secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk untuk peluncuran. Namun, Anda perlu membagikan kunci KMS yang digunakan untuk mengenkripsi snapshot yang direferensikan AMI. Untuk informasi selengkapnya, lihat [Mengizinkan organisasi dan OU untuk menggunakan kunci KMS](#).

7. Setelah selesai, pilih Simpan perubahan.
8. (Opsional) Untuk melihat organisasi atau OU yang telah Anda bagi AMI, pilih AMI dalam daftar, pilih tab Izin, dan gulir ke bawah ke Organisasi/OU bersama. Untuk mencari AMI yang dibagikan dengan Anda, lihat [Mencari AMI bersama](#).

Membagikan AMI (Alat untuk Windows) PowerShell)

Gunakan [Edit-EC2ImageAttribute](#) perintah (Alat untuk Windows PowerShell) untuk berbagi AMI seperti yang ditunjukkan pada contoh berikut.

Untuk berbagi AMI dengan organisasi atau OU

Perintah berikut memberikan izin peluncuran untuk AMI yang ditentukan ke organisasi tertentu.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -  
Attribute launchPermission -OperationType add -OrganizationArn  
"arn:aws:organizations::123456789012:organization/o-123example"
```

 Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, dan sistem secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk 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](#).

Untuk berhenti berbagi AMI dengan organisasi atau OU

Perintah berikut menghapus izin peluncuran untuk AMI yang ditentukan dari organisasi tertentu:

```
PS C:\> 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.

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission
```

Bagikan AMI (AWS CLI)

Gunakan perintah [modify-image-attribute](#) (AWS CLI) untuk berbagi AMI.

Untuk berbagi AMI dengan organisasi menggunakan AWS CLI

Perintah [modify-image-attribute](#) memberikan izin peluncuran untuk AMI yang ditentukan ke organisasi tertentu. Perhatikan bahwa Anda harus menentukan ARN lengkap, bukan hanya ID-nya.

```
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 menggunakan AWS CLI

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}]]"
```

Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, dan sistem secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk 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](#).

Berhenti berbagi AMI

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk berhenti berbagi AMI dengan organisasi atau OU.

Berhenti berbagi AMI (konsol)

Untuk berhenti berbagi AMI dengan organisasi atau OU menggunakan konsol

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 Organisasi/OU berbagi, pilih organisasi atau OU yang ingin Anda hentikan berbagi AMI, lalu pilih Hapus pilihan.
5. Setelah selesai, pilih Simpan perubahan.
6. (Opsional) Untuk mengonfirmasi bahwa Anda telah berhenti membagikan AMI dengan organisasi atau OU, pilih AMI dalam daftar, pilih tab Izin, dan gulir ke bawah ke Organisasi/OU berbagi.

Berhenti berbagi AMI (AWS CLI)

Gunakan perintah [modify-image-attribute](#) atau [reset-image-attribute](#) (AWS CLI) untuk berhenti berbagi AMI.

Untuk berhenti berbagi AMI dengan organisasi atau OU menggunakan AWS CLI

Perintah [modify-image-attribute](#) menghapus izin peluncuran untuk AMI yang ditentukan dari organisasi tertentu. Perhatikan bahwa Anda harus menentukan ARN.

```
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 menggunakan AWS CLI

Perintah [reset-image-attribute](#) akan 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
```

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.

Melihat organisasi dan OU yang berbagi AMI Anda

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk memeriksa organisasi dan OU mana yang telah Anda bagikan AMI Anda.

Melihat organisasi dan OU yang berbagi AMI Anda (konsol)

Untuk memeriksa organisasi dan OU mana yang telah Anda bagikan AMI menggunakan konsol

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 Organisasi/OU berbagi.

Untuk mencari AMI yang dibagikan dengan Anda, lihat [Mencari AMI bersama](#).

Melihat organisasi dan OU yang berbagi AMI Anda (AWS CLI)

Anda dapat memeriksa organisasi dan OU yang berbagi AMI Anda dengan menggunakan perintah [describe-image-attribute](#) (AWS CLI) dan atribut `launchPermission`.

Untuk memeriksa dengan organisasi dan OU mana Anda telah membagikan AMI Anda menggunakan AWS CLI

Perintah [describe-image-attribute](#) akan menjelaskan atribut `launchPermission` untuk AMI yang ditentukan, dan menampilkan organisasi dan OU yang berbagi AMI Anda.

```
aws ec2 describe-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

Contoh tanggapan

```
{  
  "ImageId": "ami-0abcdef1234567890",  
  "LaunchPermissions": [  
    {  
      "OrganizationalUnitArn": "arn:aws:organizations::111122223333:ou/  
o-123example/ou-1234-5example"  
    }  
  ]  
}
```

Mendapatkan ARN

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 nomor akun manajemen.

Sebelum Anda bisa mendapatkan ARN, Anda harus memiliki izin untuk menjelaskan organisasi dan unit organisasi. Kebijakan berikut ini memberikan izin yang diperlukan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "organizations:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```

Untuk mendapatkan ARN suatu organisasi

Gunakan perintah [describe-organization](#) dan parameter `--query` yang diatur ke `'Organization.Arn'` untuk menampilkan ARN organisasi saja.

```
aws organizations describe-organization --query 'Organization.Arn'
```

Contoh tanggapan

```
"arn:aws:organizations::123456789012:organization/o-123example"
```

Untuk mendapatkan ARN unit organisasi

Gunakan perintah [describe-organizational-unit](#), tentukan ID OU, dan atur parameter `--query` ke `'OrganizationalUnit.Arn'` untuk menampilkan ARN unit organisasi saja.

```
aws organizations describe-organizational-unit --organizational-unit-id ou-1234-5example --query 'OrganizationalUnit.Arn'
```

Contoh tanggapan

```
"arn:aws:organizations::123456789012:ou/o-123example/ou-1234-5example"
```

Membagikan AMI kepada akun AWS tertentu

Anda dapat berbagi AMI dengan spesifik Akun AWS tanpa membuat AMI 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 pengenalan Akun AWS](#) di Panduan Referensi AWS Account Management.

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 membagikan AMI, tag yang ditentukan pengguna tidak tersedia untuk semua Akun AWS yang digunakan bersama AMI.
- 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 mengizinkan Akun AWS untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi snapshot. Untuk informasi selengkapnya, lihat [Mengizinkan organisasi dan OU untuk menggunakan kunci KMS](#). Untuk menyiapkan kebijakan utama yang Anda perlukan untuk meluncurkan instans Auto Scaling saat menggunakan kunci terkelola pelanggan untuk enkripsi, lihat [AWS KMS key Kebijakan yang diperlukan untuk digunakan dengan volume terenkripsi di](#) Panduan Pengguna Auto Scaling 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 [Menyalin AMI](#).
- 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 [Penyalinan lintas akun](#).

- Penagihan — Anda tidak ditagih ketika AMI Anda digunakan oleh orang lain Akun AWS untuk meluncurkan instans. Akun yang meluncurkan instans menggunakan AMI akan dikenai biaya untuk instans yang diluncurkan.

Membagikan AMI (konsol)

Untuk memberikan izin peluncuran eksplisit menggunakan konsol

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 bagikan 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.

Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, sistem akan secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk untuk peluncuran. Namun, Anda perlu berbagi kunci KMS yang digunakan untuk mengenkripsi snapshot yang ditunjuk oleh AMI. Untuk informasi selengkapnya, lihat [Membagikan snapshot Amazon EBS](#) di Panduan Pengguna Amazon EBS.

7. Setelah selesai, pilih Simpan perubahan.
8. (Opsional) Untuk melihat Akun AWS ID yang telah Anda bagikan AMI, pilih AMI dalam daftar, dan pilih tab Izin. Untuk mencari AMI yang dibagikan dengan Anda, lihat [Mencari AMI bersama](#).

Membagikan AMI (Alat untuk Windows) PowerShell)

Gunakan [Edit-EC2ImageAttribute](#) perintah (Alat untuk Windows PowerShell) untuk berbagi AMI seperti yang ditunjukkan pada contoh berikut.

Untuk memberikan izin peluncuran eksplisit

Perintah berikut ini memberikan izin peluncuran untuk AMI yang ditentukan ke Akun AWS tertentu. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid, dan ganti *account-id* dengan Akun AWS ID 12 digit.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission -OperationType add -UserId "account-id"
```

Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, sistem akan secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk untuk peluncuran. Namun, Anda perlu berbagi kunci KMS yang digunakan untuk mengenkripsi snapshot yang ditunjuk oleh AMI. Untuk informasi selengkapnya, lihat [Membagikan snapshot Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Untuk menghapus izin peluncuran bagi sebuah akun

Perintah berikut menghapus izin peluncuran untuk AMI yang ditentukan dari Akun AWS tertentu. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid, dan ganti *account-id* dengan Akun AWS ID 12 digit.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission -OperationType remove -UserId "account-id"
```

Untuk menghapus semua izin peluncuran

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. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid.

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute  
launchPermission
```

Membagikan AMI (AWS CLI)

Gunakan perintah [modify-image-attribute](#) (AWS CLI) untuk berbagi AMI seperti yang ditunjukkan pada contoh berikut.

Untuk memberikan izin peluncuran eksplisit

Perintah berikut ini memberikan izin peluncuran untuk AMI yang ditentukan ke Akun AWS tertentu. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid, dan ganti *account-id* dengan Akun AWS ID 12 digit.

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Add=[{UserId=account-id}]"
```

Note

Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk oleh AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, sistem akan secara otomatis menyediakan akses kepada instans menuju snapshot Amazon EBS yang ditunjuk untuk peluncuran. Namun, Anda perlu berbagi kunci KMS yang digunakan untuk mengenkripsi snapshot yang ditunjuk oleh AMI. Untuk informasi selengkapnya, lihat [Membagikan snapshot Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Untuk menghapus izin peluncuran bagi sebuah akun

Perintah berikut menghapus izin peluncuran untuk AMI yang ditentukan dari Akun AWS tertentu. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid, dan ganti *account-id* dengan Akun AWS ID 12 digit.

```
aws ec2 modify-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --launch-permission "Remove=[{UserId=account-id}]"
```

Untuk menghapus semua izin peluncuran

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. Pada contoh berikut, ganti contoh ID AMI dengan ID AMI yang valid.

```
aws ec2 reset-image-attribute \  
  --image-id ami-0abcdef1234567890 \  
  --attribute launchPermission
```

Batalkan AMI yang dibagikan 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 ingin lagi 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 akan Akun AWS dihapus dari izin peluncuran untuk AMI yang ditentukan.

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](#)
- [Mencari AMI yang dibagikan ke akun Anda](#)

Batasan

- Anda dapat menghapus akun Anda dari izin peluncuran AMI yang dibagikan Akun AWS hanya dengan Anda. Anda tidak dapat menggunakan `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. Baik dengan menyebutkan Wilayah dalam perintah, atau menggunakan [variabel lingkungan](#) `AWS_DEFAULT_REGION`.
- Hanya SDK AWS CLI dan SDK 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 AMI yang dibagikan dengan Anda Akun AWS

Gunakan perintah [cancel-image-launch-permission](#) dan sebutkan ID AMI.

```
aws ec2 cancel-image-launch-permission \  
  --image-id ami-0123456789example \  
  --region us-east-1
```

Output yang diharapkan

```
{  
  "Return": true  
}
```

PowerShell

Untuk membatalkan AMI yang dibagikan dengan Anda Akun AWS menggunakan AWS Tools for PowerShell

Gunakan perintah [Stop-EC2ImageLaunchPermission](#) dan sebutkan ID AMI.

```
Stop-EC2ImageLaunchPermission \  
  -ImageId ami-0123456789example \  
  -Region us-east-1
```

Output yang diharapkan

```
True
```

Mencari AMI yang dibagikan ke akun Anda

Untuk menemukan AMI yang dibagikan dengan Anda Akun AWS, lihat [Mencari AMI bersama](#).

Menggunakan bookmark

Jika Anda telah membuat AMI publik, atau membagikan AMI dengan yang lain Akun AWS, Anda dapat membuat bookmark yang memungkinkan pengguna mengakses AMI Anda dan segera meluncurkan instance di akun mereka sendiri. Ini adalah cara mudah untuk berbagi referensi AMI sehingga pengguna tidak perlu menghabiskan waktu mencari AMI Anda untuk menggunakannya.

Perhatikan bahwa AMI harus bersifat publik, atau Anda harus membagikannya kepada pengguna yang ingin Anda kirim bookmark.

Untuk membuat bookmark untuk AMI Anda

1. Ketikkan URL dengan informasi berikut, di mana wilayah adalah Wilayah tempat AMI Anda berada:

```
https://console.aws.amazon.com/ec2/v2/home?  
region=region#LaunchInstanceWizard:ami=ami_id
```

Misalnya, URL ini meluncurkan instans dari AMI ami-0abcdef1234567890 di Wilayah us-east-1 AS Timur (Virginia Utara):

```
https://console.aws.amazon.com/ec2/v2/home?region=us-  
east-1#LaunchInstanceWizard:ami=ami-0abcdef1234567890
```

2. Menyebarkan tautan kepada pengguna yang ingin menggunakan AMI Anda.
3. Untuk menggunakan bookmark, pilih tautan atau salin dan tempel ke peramban Anda. Wizard peluncuran akan terbuka, dengan AMI yang sudah dipilih.

Panduan untuk 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

- [Perbarui alat AMI sebelum menggunakannya](#)
- [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 \(opsional\)](#)
- [Lindungi diri Anda](#)

Jika Anda membangun AMI AWS Marketplace, lihat [Praktik terbaik untuk membangun AMI](#) di Panduan AWS Marketplace Penjual untuk pedoman, kebijakan, dan praktik terbaik.

Untuk informasi tambahan tentang berbagi AMI dengan aman, lihat artikel berikut ini:

- [Cara Membagikan dan Menggunakan AMI Publik dengan Aman](#)
- [Memublikasikan AMI publik: Persyaratan Penguatan dan Pembersihan](#)

Perbarui alat AMI sebelum menggunakannya

Untuk AMI yang didukung oleh penyimpanan instans, kami sarankan mengunduh dan meningkatkan alat pembuatan Amazon EC2 AMI sebelum menggunakannya. Ini memastikan bahwa AMI yang baru berdasarkan AMI bersama Anda memiliki alat AMI terbaru.

Untuk [Amazon Linux 2](#), instal paket `aws-amitools-ec2` dan tambahkan alat AMI ke Jalur Anda dengan perintah berikut. Untuk [AMI Amazon Linux](#), paket `aws-amitools-ec2` telah diinstal secara default.

```
[ec2-user ~]$ sudo yum install -y aws-amitools-ec2 && export PATH=$PATH:/opt/aws/bin  
> /etc/profile.d/aws-amitools-ec2.sh && . /etc/profile.d/aws-amitools-ec2.sh
```

Tingkatkan alat AMI dengan perintah berikut:

```
[ec2-user ~]$ sudo yum upgrade -y aws-amitools-ec2
```

Untuk distribusi lainnya, pastikan Anda memiliki alat AMI terbaru.

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**, tidak boleh 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 [dokumentasi 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 penyalakan 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 tautan lokal dan hanya valid dari instans.

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.

Lindungi diri Anda

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. Khususnya, kecualikan semua pasangan kunci publik/privat SSH milik pengguna dan file SSH `authorized_keys` saat memaketkan 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 keculikan 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).

AMI berbayar

AMI berbayar adalah AMI yang terdaftar untuk dijual di AWS Marketplace. AWS Marketplace Ini 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 disusun 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 [AMI bersama](#).

Meluncurkan instans dari AMI berbayar sama seperti meluncurkan instans dari AMI lainnya. Tidak perlu parameter tambahan. Instans dibebankan 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 menjadi platform e-commerce tunggal terpadu untuk menjual perangkat lunak dan layanan melalui AWS. Untuk informasi tentang cara menyebarkan dan menjual perangkat lunak AWS Marketplace, lihat [Menjual di AWS Marketplace](#). AWS Marketplace mendukung AMI yang didukung oleh Amazon EBS.

Daftar Isi

- [Menjual AMI Anda](#)
- [Mencari AMI berbayar](#)
- [Membeli AMI berbayar](#)
- [Mendapatkan kode produk untuk instans Anda](#)
- [Menggunakan dukungan berbayar](#)
- [Tagihan untuk AMI berbayar dan didukung](#)
- [Kelola AWS Marketplace langganan Anda](#)

Menjual AMI Anda

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

Untuk informasi tentang cara menjual AMI Anda di AWS Marketplace, lihat [Menjual di AWS Marketplace](#).

Mencari AMI berbayar

Ada beberapa cara Anda dapat mencari AMI yang tersedia untuk dibeli. Misalnya, Anda dapat menggunakan [AWS Marketplace](#), konsol Amazon EC2, atau baris perintah. Selain itu, developer mungkin akan memberi tahu Anda tentang AMI berbayar.

Mencari AMI berbayar menggunakan konsol

Untuk menemukan AMI berbayar menggunakan konsol

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. Di bilah Pencarian, pilih Alias pemilik, lalu =, lalu aws-marketplace.
5. Jika Anda mengetahui kode produk, pilih Kode Produk, lalu =, lalu masukkan kode produk.

Temukan AMI berbayar menggunakan AWS Marketplace

Untuk menemukan AMI berbayar menggunakan AWS Marketplace

1. Buka [AWS Marketplace](#).
2. Masukkan nama sistem operasi di bidang pencarian, lalu pilih tombol pencarian (kaca pembesar).
3. Untuk mempersempit hasil lebih lanjut, gunakan salah satu kategori atau filter.
4. Setiap produk diberi label dengan tipe produk: baik AMI maupun Software as a Service.

Temukan AMI berbayar menggunakan AWS CLI

Anda dapat menemukan AMI berbayar menggunakan perintah [describe-images](#) (AWS CLI).

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

Perintah ini memberikan berbagai detail yang menjelaskan setiap AMI, termasuk kode produk untuk AMI berbayar. Output dari `describe-images` mencakup entri untuk kode produk seperti berikut:

```
"ProductCodes": [
  {
    "ProductCodeId": "product_code",
    "ProductCodeType": "marketplace"
  }
],
```

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
  --owners aws-marketplace \
  --filters "Name=product-code,Values=product_code" \
  --query "sort_by(Images, &CreationDate)[-1].[ImageId]"
```

Temukan AMI berbayar menggunakan Alat untuk Windows PowerShell

Anda dapat menemukan AMI berbayar menggunakan [Get-EC2Image](#) perintah berikut.

```
PS C:\> Get-EC2Image -Owner aws-marketplace
```

Output untuk AMI berbayar mencakup kode produk.

ProductCodeId	ProductCodeType
-----	-----
<i>product_code</i>	marketplace

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

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

Membeli AMI berbayar

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

Biasanya, penjual AMI berbayar memberi Anda informasi tentang AMI, termasuk harga dan tautan tempat Anda dapat membelinya. Ketika Anda mengklik 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 sebuah AWS Marketplace instance](#).

Berlangganan produk menggunakan AWS Marketplace

Untuk menggunakannya AWS Marketplace, Anda harus memiliki AWS akun. Untuk meluncurkan instans dari AWS Marketplace produk, Anda harus mendaftar untuk menggunakan layanan Amazon EC2, dan Anda harus berlangganan produk untuk meluncurkan instans. Ada dua cara untuk berlangganan produk di AWS Marketplace:

- AWS Marketplace situs web: Anda dapat meluncurkan perangkat lunak yang telah dikonfigurasi sebelumnya dengan cepat dengan fitur penyebaran 1-Klik.
- Wizard peluncuran Amazon EC2: Anda dapat mencari AMI dan meluncurkan instans langsung dari wizard. Untuk informasi selengkapnya, lihat [Luncurkan sebuah AWS Marketplace instance](#).

Mendapatkan kode produk untuk instans Anda

Anda dapat mengambil kode AWS Marketplace produk untuk instance Anda menggunakan metadata instance-nya. Jika instans memiliki kode produk, Amazon EC2 akan memberikannya. Untuk informasi selengkapnya tentang pengambilan metadata instans, lihat [Mengambil metadata instans](#).

Untuk mengambil kode produk, gunakan perintah untuk sistem operasi instans Anda.

Linux

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/product-codes
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/product-codes
```

Windows

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/product-codes
```

Menggunakan dukungan berbayar

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.

Important

Anda tidak dapat menggunakan produk dukungan dengan Instans Tercadang. Anda selalu membayar harga yang ditetapkan oleh penjual produk dukungan.

Untuk mengaitkan kode produk dengan AMI Anda, gunakan salah satu perintah berikut, di mana `ami_id` adalah ID AMI dan `product_code` adalah kode produk:

- [modify-image-attribute](#) (AWS CLI)

```
aws ec2 modify-image-attribute --image-id ami_id --product-codes "product_code"
```

- [Edit-EC2ImageAttribute](#) (AWS Tools for Windows PowerShell)

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami_id -ProductCode product_code
```

Setelah Anda mengatur atribut kode produk, atribut tidak dapat diubah atau dihapus.

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 AWS Marketplace langganan Anda

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

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 terhubung ke instans yang berjalan.

Untuk membatalkan AWS Marketplace langganan

1. Pastikan Anda telah mengakhiri instans yang berjalan dari langganan.
 - a. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
 - b. Di panel navigasi, pilih Instans.
 - c. Pilih instans, lalu pilih Status instans, Akhiri instans.
 - d. Pilih Akhiri saat diminta untuk mengonfirmasi.
2. Masuk ke [AWS Marketplace](#), dan pilih Akun Marketplace Anda, lalu Kelola langganan perangkat lunak Anda.
3. Pilih Batalan langganan. Anda diminta untuk mengonfirmasi pembatalan.

Note

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 AWS Marketplace situs web, atau melalui panduan peluncuran di konsol Amazon EC2.

Siklus hidup AMI

Anda dapat membuat AMI Anda sendiri, menyalinnya, mencadangkannya, dan memeliharanya sampai Anda siap untuk membuatnya usang atau membatalkan pendaftarannya.

Daftar Isi

- [Membuat AMI](#)
- [Memodifikasi AMI](#)
- [Menyalin AMI](#)
- [Simpan dan pulihkan AMI menggunakan S3](#)
- [Membuat usang sebuah AMI](#)
- [Menonaktifkan AMI](#)
- [Mengarsipkan snapshot AMI](#)
- [Deregister \(hapus\) AMI](#)
- [Otomatisasi siklus hidup AMI yang didukung EBS](#)

Membuat AMI

Anda dapat membuat AMI Linux atau Windows yang didukung oleh volume Amazon EBS. Anda juga dapat membuat AMI Linux yang didukung oleh volume penyimpanan instance (AMI Windows tidak mendukung penyimpanan instance untuk perangkat root). Anda juga dapat menggunakan Windows Sysprep untuk membuat AMI Windows.

Topik

- [Buat AMI yang didukung Amazon EBS-Backed](#)
- [Buat AMI Linux yang didukung penyimpanan instans](#)
- [Buat AMI dengan Windows Sysprep](#)

Buat AMI yang didukung Amazon EBS-Backed

Untuk membuat AMI yang didukung Amazon EBS-backed, mulailah dari instance yang telah Anda luncurkan dari AMI yang didukung Amazon EBS-backed. Ini bisa berupa AMI yang Anda peroleh dari AWS Marketplace, AMI yang telah Anda buat menggunakan [AWS Server Migration Service](#) atau [VM Impor/Ekspor](#), atau AMI lain yang dapat Anda akses. Setelah Anda mengustomisasi instans sesuai

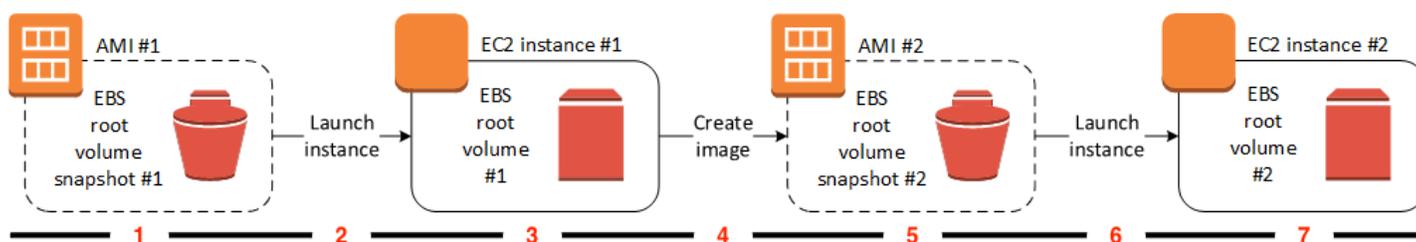
kebutuhan, buat dan daftarkan AMI baru tersebut, yang dapat Anda gunakan untuk meluncurkan instans baru dengan kustomisasi ini.

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 AMI yang didukung penyimpanan instans. Untuk informasi selengkapnya tentang perbedaan antara instans yang didukung oleh Amazon EBS dan instans yang didukung penyimpanan instans, serta cara menentukan tipe perangkat root untuk instans Anda, lihat [Penyimpanan untuk perangkat root](#). Untuk informasi tentang membuat instance store-backed AMI, lihat [Buat AMI Linux yang didukung penyimpanan instans](#)

Ikhtisar pembuatan AMI yang didukung Amazon EBS

Diagram berikut merangkum proses pembuatan AMI yang didukung Amazon EBS dari instans EC2 yang sedang berjalan: Mulai dengan AMI yang ada, luncurkan instans, kustomisasi, buat AMI baru darinya, dan terakhir luncurkan instans 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 Anda peroleh dari AWS Marketplace, AMI yang telah Anda buat menggunakan [AWS Server Migration Service](#) atau [VM Impor/Ekspor](#), atau AMI lain yang dapat Anda akses. Anda akan mengustomisasi AMI ini untuk kebutuhan Anda.

Dalam diagram, Snapshot volume root EBS #1 menunjukkan bahwa AMI adalah AMI yang didukung Amazon EBS 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 proses pembuatan AMI, Amazon EC2 membuat snapshot volume root instans 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.

Bergantung pada ukuran volume, pembuatan AMI dapat memakan waktu beberapa menit untuk selesai (terkadang 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 ke volume perangkat root, pemetaan perangkat blok untuk AMI yang baru akan berisi informasi untuk volume tersebut.

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 ke instans Anda saat Anda mengustomisasi instans, pemetaan perangkat blok untuk AMI yang baru akan 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 [Pemetaan perangkat blok](#).

Saat membuat instans baru dari AMI yang didukung EBS, Anda harus menginisialisasi volume root dan penyimpanan EBS tambahan sebelum memasukkannya ke tahap produksi. Untuk informasi selengkapnya, lihat [Menginisialisasi volume Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Buat AMI dari sebuah instance

Anda dapat membuat AMI menggunakan AWS Management Console atau baris perintah.

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 instans yang didukung Amazon EBS.

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 Tidak melakukan boot ulang, biarkan kotak centang Aktifkan kosong (default), atau pilih.
 - Jika kotak centang Aktifkan untuk Tidak ada boot ulang dikosongkan, saat Amazon EC2 membuat AMI baru, AMI akan melakukan boot ulang instans sehingga dapat mengambil snapshot dari volume yang dilampirkan ketika data dalam keadaan diam untuk memastikan status yang konsisten.
 - Jika kotak centang Aktifkan untuk Tidak ada boot ulang dipilih, saat Amazon EC2 membuat AMI baru, AMI tidak mematikan dan melakukan boot ulang instans.
 - 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

 Warning

Jika Anda memilih untuk mengaktifkan Tidak melakukan boot ulang, kami tidak dapat menjamin integritas sistem file dari gambar yang dibuat.

- 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](#). 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. 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.

- f. 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 instans dari AMI ini, Amazon EC2 menggunakan snapshot untuk membuat volume perangkat root.

AWS CLI

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [create-image](#) (AWS CLI)
- [New-EC2Image](#) (AWS Tools for Windows PowerShell)

Membuat AMI Linux dari snapshot

Jika Anda memiliki snapshot volume perangkat root dari sebuah instance, Anda dapat membuat AMI Linux dari snapshot ini menggunakan AWS Management Console atau baris perintah. Fitur ini saat ini tidak tersedia untuk instance Windows.

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 perangkat root. Untuk informasi selengkapnya, lihat [Nama perangkat di instans Amazon EC2](#).
 - e. Untuk Tipe virtualisasi, pilih tipe virtualisasi yang akan digunakan oleh instans yang diluncurkan dari AMI ini. Untuk informasi selengkapnya, lihat [Tipe virtualisasi AMI](#).

- f. (Hanya untuk virtualisasi paravirtual) Untuk ID Kernel, pilih kernel sistem operasi untuk gambar tersebut. Jika Anda menggunakan snapshot volume perangkat root suatu instans, pilih ID kernel yang sama dengan instans asli. 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 booting dengan mode boot yang didukung oleh jenis instance. Untuk informasi selengkapnya, lihat [Mengatur mode boot sebuah AMI](#).
- i. (Opsional) Di bawah Blokir 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 beberapa 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.

AWS CLI

Untuk membuat AMI dari snapshot menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [daftar-gambar \(CLI\)](#) AWS
- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

Luncurkan instance dari AMI yang Anda buat

Anda dapat meluncurkan instans dari AMI yang Anda buat dari instans atau snapshot.

Untuk meluncurkan instans dari AMI Anda

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, di bagian Gambar, pilih AMI.
3. Atur filter ke Dimiliki oleh saya dan pilih AMI Anda.
4. Pilih Luncurkan instans dari AMI.
5. Terima nilai default atau tentukan nilai kustom di wizard peluncuran instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Buat AMI Linux yang didukung penyimpanan instans

AMI yang Anda tentukan saat meluncurkan instans akan menentukan tipe volume perangkat root.

Untuk membuat AMI Linux yang didukung penyimpanan instans, mulai dari instans yang telah Anda luncurkan dari AMI Linux yang sudah ada yang didukung oleh penyimpanan instans. 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 Windows AMI yang didukung instance-store karena AMI Windows tidak mendukung penyimpanan instance untuk perangkat root.

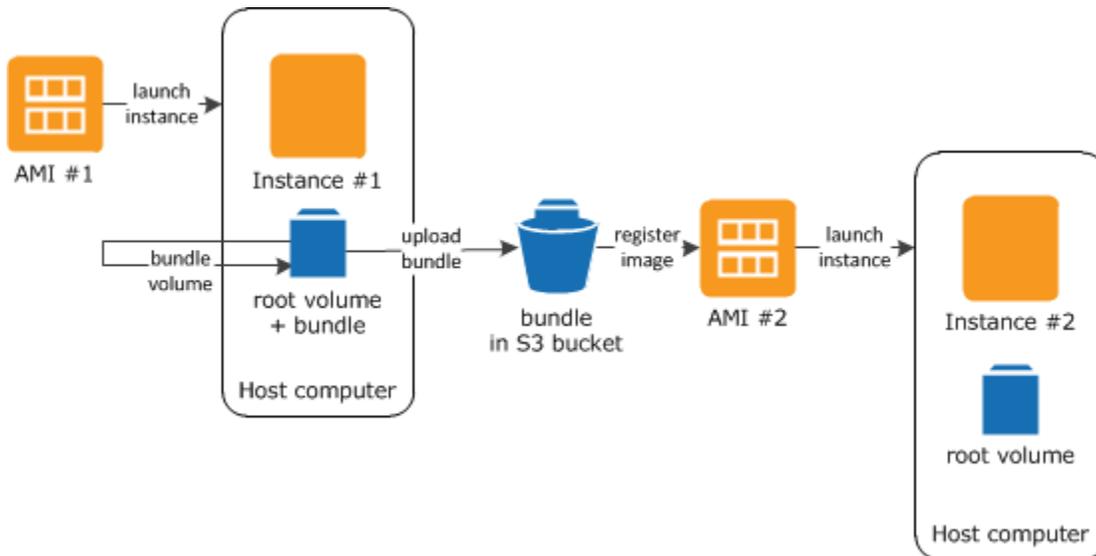
Important

Hanya jenis instance berikut yang mendukung volume penyimpanan instance sebagai perangkat root: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Proses pembuatan AMI berbeda untuk AMI yang didukung Amazon EBS. Untuk informasi selengkapnya tentang perbedaan antara instans yang didukung oleh Amazon EBS dan instans yang didukung penyimpanan instans, serta cara menentukan tipe perangkat root untuk instans Anda, lihat [Penyimpanan untuk perangkat root](#). Jika Anda perlu membuat AMI yang didukung Amazon EBS-backed, lihat [Buat AMI yang didukung Amazon EBS-Backed](#)

Ikhtisar proses pembuatan untuk AMI yang didukung penyimpanan instans

Diagram berikut meringkas proses pembuatan AMI dari instans yang didukung penyimpanan instans.



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 paket. Perlu beberapa menit untuk menyelesaikan proses pembuatan paket. Setelah proses selesai, Anda akan memiliki paket, yang terdiri atas manifes gambar (`image.manifest.xml`) dan file (`image.part.xx`) yang berisi templat untuk volume root. Berikutnya, Anda mengunggah paket tersebut ke bucket Amazon S3 kemudian mendaftarkan AMI Anda.

Note

Untuk mengunggah objek ke bucket S3 untuk AMI Linux yang didukung penyimpanan instans 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 [Mengendalikan kepemilikan objek yang diunggah menggunakan Kepemilikan Objek S3](#).

Saat Anda meluncurkan instans menggunakan AMI yang baru, kami membuat volume root untuk instans tersebut menggunakan paket yang Anda unggah ke Amazon S3. Ruang penyimpanan yang digunakan oleh paket di Amazon S3 akan menimbulkan biaya sampai Anda menghapusnya. Untuk informasi selengkapnya, lihat [Deregister \(hapus\) AMI](#).

Jika Anda menambahkan volume penyimpanan instans ke instans Anda selain ke volume perangkat root, pemetaan perangkat blok untuk AMI baru berisi informasi untuk volume ini, dan pemetaan

perangkat blok untuk instans yang Anda luncurkan dari AMI yang baru akan secara otomatis berisi informasi untuk volume ini. Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#).

Prasyarat

Sebelum dapat membuat AMI, Anda harus menyelesaikan tugas berikut:

- Menginstal alat AMI. Untuk informasi selengkapnya, lihat [Mengatur alat AMI](#).
- Instal AWS CLI. Untuk informasi selengkapnya, lihat [Menyiapkan dengan AWS Command Line Interface](#).
- 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 AWS Management Console, buka konsol Amazon S3 [di https://console.aws.amazon.com/s3/](https://console.aws.amazon.com/s3/) dan pilih Create 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 Anda memiliki ID AWS akun Anda. Untuk informasi selengkapnya, lihat [Melihat Akun AWS pengenal](#) di Panduan Referensi Manajemen AWS Akun.
- Pastikan Anda memiliki kredensial untuk menggunakan AWS CLI. Untuk informasi selengkapnya, lihat [Praktik Terbaik untuk AWS akun](#) di Panduan AWS Account Management Referensi.
- Pastikan Anda memiliki sertifikat X.509 dan kunci privat yang sesuai.
 - Jika Anda perlu membuat sertifikat X.509, lihat [Mengelola sertifikat penandatanganan](#). Sertifikat X.509 dan kunci privat digunakan untuk mengenkripsi dan mendekripsi AMI Anda.
 - [Tiongkok (Beijing)] Gunakan sertifikat `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-cn-north-1.pem`.
 - [AWS GovCloud (AS-Barat)] Gunakan `$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.

Tugas

- [Mengatur alat AMI](#)

- [Membuat AMI dari instans Amazon Linux yang didukung penyimpanan instans](#)
- [Membuat AMI dari instans Ubuntu yang didukung penyimpanan instans](#)
- [Konversikan AMI yang didukung penyimpanan instans menjadi AMI yang didukung Amazon EBS](#)

Mengatur alat AMI

Gunakan alat AMI untuk membuat dan mengelola AMI Linux yang didukung penyimpanan instans. 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 terverifikasi, 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 tersebut -- ec2/amitools/version (LoadError)”, selesaikan langkah berikutnya untuk menambahkan lokasi instalasi alat AMI Anda ke jalur Anda. RUBYLIB

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*, di mana *x.x.x* adalah nomor versi alat (misalnya, ec2-ami-tools-1.5.7).

4. Atur variabel lingkungan EC2_AMIT00L_HOME untuk direktori instalasi alat. Sebagai contoh:

```
[ec2-user ~]$ export EC2_AMIT00L_HOME=/usr/local/ec2/ec2-ami-tools-x.x.x
```

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

```
[ec2-user ~]$ export PATH=$EC2_AMIT00L_HOME/bin:$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 di alat AMI memerlukan penandatanganan sertifikat (juga dikenal sebagai sertifikat X.509). 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-signing-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 penandatanganan bagi pengguna, gunakan perintah [update-signing-certificate](#). Perintah berikut ini menonaktifkan sertifikat:

```
aws iam update-signing-certificate --certificate-id OFHPLP4ZULTHYPMSYEX704BEXAMPLE --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 OFHPLP4ZULTHYPMSYEX704BEXAMPLE
```

Membuat AMI dari instans yang didukung penyimpanan instans

Prosedur berikut ini adalah untuk membuat AMI yang didukung penyimpanan instans dari instans yang didukung penyimpanan instans. Sebelum memulai, pastikan Anda telah membaca [Prasyarat](#).

Topik

- [Membuat AMI dari instans Amazon Linux yang didukung penyimpanan instans](#)
- [Membuat AMI dari instans Ubuntu yang didukung penyimpanan instans](#)

Membuat AMI dari instans Amazon Linux yang didukung penyimpanan instans

Bagian ini menjelaskan pembuatan AMI dari instans Amazon Linux. Prosedur berikut mungkin tidak berfungsi untuk instans yang menjalankan distribusi Linux lainnya. Untuk prosedur spesifik Ubuntu, lihat [Membuat AMI dari instans Ubuntu yang didukung penyimpanan instans](#).

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
```

Untuk membuat AMI dari instans Amazon Linux yang didukung penyimpanan instans

Prosedur ini mengasumsikan bahwa Anda telah memenuhi prasyarat dalam [Prasyarat](#).

Dalam perintah berikut, ganti setiap *placeholder input pengguna* 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 sertifikat X.509 dan kunci privat terkait dari komputer Anda ke direktori `/tmp/cert` pada instans dengan alat penyalin aman, seperti [scp](#). Opsi `-i my-private-key.pem` dalam perintah `scp` berikut adalah kunci privat yang Anda gunakan untuk terhubung ke instan Anda dengan SSH, bukan kunci privat X.509. Sebagai contoh:

```
you@your_computer:~ $ scp -i my-private-key.pem /  
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem /
```

```
path/to/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717 0.7KB/s 00:00
cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.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 instance memiliki penyimpanan sementara yang terpasang pada `/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-HKZYKTAIG2ECMXIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-HKZYKTAIG2ECMXIBH3HXV4ZBEXAMPLE.pem -u 123456789012 -r x86_64 -e /tmp/cert --partition gpt
```

 Note

Untuk Wilayah China (Beijing) dan AWS GovCloud (AS-Barat), gunakan `--ec2cert` parameter dan tentukan sertifikat sesuai [prasyarat](#).

Perlu waktu beberapa menit untuk membuat gambar. Setelah perintah ini selesai, direktori /tmp Anda (atau direktori non-default) memuat paketan (`image.manifest.xml`, ditambah beberapa file `xximage.part.`).

- 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 [Pemetaan perangkat blok](#).

- 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 my-s3-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 my-s3-  
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.

Membuat AMI dari instans Ubuntu yang didukung penyimpanan instans

Bagian ini menjelaskan pembuatan AMI dari instans Ubuntu Linux dengan volume penyimpanan instans sebagai volume root. Prosedur berikut mungkin tidak berfungsi untuk instans yang menjalankan distribusi Linux lainnya. Untuk prosedur khusus Amazon Linux, lihat [Membuat AMI dari instans Amazon Linux yang didukung penyimpanan instans](#).

Untuk mempersiapkan penggunaan alat AMI (khusus instans HVM)

Alat AMI memerlukan GRUB Legacy untuk melakukan boot dengan benar. Namun, Ubuntu dikonfigurasi untuk menggunakan GRUB 2. Anda harus memeriksa untuk memastikan bahwa instans Anda menggunakan GRUB Legacy, dan jika tidak, Anda perlu menginstal dan mengonfigurasinya.

Instans HVM juga memerlukan alat pembuat partisi diinstal agar alat AMI dapat berfungsi dengan baik.

1. GRUB Legacy (versi 0.9x atau kurang) harus diinstal di instans Anda. Periksa untuk mengetahui keberadaan GRUB Legacy dan instal jika perlu.
 - a. Periksa versi instalasi GRUB Anda.

```
ubuntu:~$ grub-install --version
grub-install (GRUB) 1.99-21ubuntu3.10
```

Dalam contoh ini, versi GRUB lebih besar dari 0.9x, jadi Anda harus menginstal GRUB Legacy. Lanjut ke [Step 1.b](#). Jika GRUB Legacy sudah ada, Anda dapat melewati ke [Step 2](#).

- b. Instal paket grub menggunakan perintah berikut.

```
ubuntu:~$ sudo apt-get install -y grub
```

2. Instal paket manajemen partisi berikut dengan menggunakan manajer paket untuk distribusi Anda.
 - gdisk (beberapa distribusi dapat memanggil paket gptfdisk ini sebagai gantinya)
 - kpartx
 - parted

Gunakan perintah berikut.

```
ubuntu:~$ sudo apt-get install -y gdisk kpartx parted
```

3. Periksa parameter kernel untuk instans Anda.

```
ubuntu:~$ cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-3.2.0-54-virtual root=UUID=4f392932-ed93-4f8f-
aee7-72bc5bb6ca9d ro console=ttyS0 xen_emul_unplug=unnecessary
```

Perhatikan opsi yang mengikuti parameter kernel dan perangkat root: ro, console=ttyS0, dan xen_emul_unplug=unnecessary. Opsi Anda mungkin berbeda.

4. Periksa entri kernel di /boot/grub/menu.lst.

```
ubuntu:~$ grep ^kernel /boot/grub/menu.lst
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro console=hvc0
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro single
```

```
kernel /boot/memtest86+.bin
```

Perhatikan bahwa parameter `console` menunjuk `hvc0` alih-alih `ttyS0` dan bahwa parameter `xen_emul_unplug=unnecessary` tidak ada. Sekali lagi, opsi Anda mungkin berbeda.

- Ubah file `/boot/grub/menu.lst` dengan editor teks favorit Anda (seperti `vim` atau `nano`) untuk mengubah konsol dan menambahkan parameter yang Anda identifikasi sebelumnya ke entri `boot`.

```
title          Ubuntu 12.04.3 LTS, kernel 3.2.0-54-virtual
root           (hd0)
kernel         /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs
ro console=ttyS0 xen_emul_unplug=unnecessary
initrd        /boot/initrd.img-3.2.0-54-virtual

title          Ubuntu 12.04.3 LTS, kernel 3.2.0-54-virtual (recovery mode)
root           (hd0)
kernel         /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro
single console=ttyS0 xen_emul_unplug=unnecessary
initrd        /boot/initrd.img-3.2.0-54-virtual

title          Ubuntu 12.04.3 LTS, memtest86+
root           (hd0)
kernel         /boot/memtest86+.bin
```

- Verifikasikan bahwa entri kernel Anda kini berisi parameter yang benar.

```
ubuntu:~$ grep ^kernel /boot/grub/menu.lst
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro console=ttyS0
xen_emul_unplug=unnecessary
kernel /boot/vmlinuz-3.2.0-54-virtual root=LABEL=cloudimg-rootfs ro single
console=ttyS0 xen_emul_unplug=unnecessary
kernel /boot/memtest86+.bin
```

- [Hanya untuk Ubuntu 14.04 dan yang lebih baru] Dimulai dengan Ubuntu 14.04, AMI Ubuntu yang didukung penyimpanan instans menggunakan tabel partisi GPT dan partisi EFI terpisah yang dipasang di `/boot/efi`. Perintah `ec2-bundle-vol` tidak akan membuat paketan partisi boot ini, sehingga Anda perlu menonaktifkan entri `/etc/fstab` dengan mengubahnya menjadi komentar untuk partisi EFI seperti yang ditunjukkan dalam contoh berikut.

```
LABEL=cloudimg-rootfs /          ext4  defaults    0 0
#LABEL=UEFI          /boot/efi      vfat  defaults    0 0
```

```
/dev/xvdb    /mnt    auto    defaults,nobootwait,comment=cloudconfig 0    2
```

Untuk membuat AMI dari instans Ubuntu yang didukung penyimpanan instans

Prosedur ini mengasumsikan bahwa Anda telah memenuhi prasyarat dalam [Prasyarat](#).

Dalam perintah berikut, ganti setiap *placeholder input pengguna* 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:

```
ubuntu:~$ mkdir /tmp/cert
```

Ini memungkinkan Anda mengecualikan kredensial Anda dari gambar yang dibuat.

- b. Salin sertifikat X.509 dan kunci privat dari komputer Anda ke direktori /tmp/cert pada instans dengan alat penyalin aman, seperti [scp](#). Opsi `-i my-private-key.pem` dalam perintah scp berikut ini adalah kunci privat yang Anda gunakan untuk terhubung ke instans Anda dengan SSH, bukan kunci privat X.509. Sebagai contoh:

```
you@your_computer:~ $ scp -i my-private-key.pem /
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem /
path/to/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717    0.7KB/s    00:00
cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 685    0.7KB/s    00:00
```

Atau, karena ini adalah file teks biasa, Anda dapat membuka sertifikat dan kunci 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 instans Anda. Pastikan Anda 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*`, `*.pgp`, `*.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 instance memiliki penyimpanan sementara yang terpasang pada `/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 kebutuhan 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.

```
ubuntu:~$ 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@ubuntu:~#
```

- b. Untuk membuat paketan AMI, jalankan perintah [ec2-bundle-vol](#) sebagai berikut.

```
root@ubuntu:~# ec2-bundle-vol -k /tmp/cert/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -u your_aws_account_id -r x86_64 -e /tmp/cert --partition gpt
```

⚠ Important

Untuk instans HVM Ubuntu 14.04 dan yang lebih baru, tambahkan tanda `--partition mbr` untuk memaketkan petunjuk boot dengan benar; jika tidak, AMI yang baru Anda buat tidak akan boot.

Perlu waktu beberapa menit untuk membuat gambar. Setelah perintah selesai, direktori `tmp` Anda berisi paketan (`image.manifest.xml`, ditambah beberapa file `xximage.part.`).

- c. Keluar dari shell root.

```
root@ubuntu:# exit
```

3. (Opsional) Untuk menambahkan volume penyimpanan instans, edit pemetaan perangkat blok dalam file `image.manifest.xml` untuk AMI Anda. Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#).

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

```
ubuntu:~$ sudo cp /tmp/image.manifest.xml /tmp/image.manifest.xml.bak
```

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

```
ubuntu:~$ sudo xmllint --format /tmp/image.manifest.xml.bak > /tmp/  
image.manifest.xml
```

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

```
<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 keluar dari editor teks Anda.
4. Untuk mengunggah paketan Anda ke Amazon S3, jalankan perintah [ec2-upload-bundle](#) sebagai berikut.

```

ubuntu:~$ ec2-upload-bundle -b my-s3-bucket/bundle_folder/bundle_name -m /tmp/
image.manifest.xml -a your_access_key_id -s your_secret_access_key

```

Important

Jika Anda berniat 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 tersebut.

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

```

ubuntu:~$ 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 yang sama di bawah ini, alih-alih `/tmp`.

6. Untuk mendaftarkan AMI Anda, jalankan perintah [register-image](#) AWS CLI sebagai berikut.

```

ubuntu:~$ aws ec2 register-image --image-location my-s3-
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.

7. [Ubuntu 14.04 dan yang lebih baru] Hapus komentar entri EFI di `/etc/fstab`; jika tidak, instans yang Anda jalankan tidak akan dapat boot ulang.

Konversikan AMI yang didukung penyimpanan instans menjadi AMI yang didukung Amazon EBS

Anda dapat mengonversi AMI Linux yang didukung penyimpanan instans yang Anda miliki ke AMI Linux yang didukung Amazon EBS.

⚠ Important

Anda tidak dapat mengonversi AMI yang tidak Anda miliki.

Untuk mengonversi AMI yang didukung penyimpanan instans ke AMI yang didukung Amazon EBS

1. Luncurkan instans Amazon Linux dari AMI yang didukung Amazon EBS. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#). Instans Amazon Linux memiliki alat AWS CLI dan AMI yang sudah diinstal sebelumnya.
2. Unggah kunci privat X.509 yang Anda gunakan untuk memaketkan AMI yang didukung penyimpanan instans ke instans Anda. Kami menggunakan kunci ini untuk memastikan hanya Anda dan Amazon EC2 yang dapat mengakses AMI.
 - a. Buat direktori sementara pada instans Anda untuk kunci privat X.509 sebagai berikut:

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

- b. Salin kunci privat X.509 dari komputer Anda ke direktori `/tmp/cert` pada instans, menggunakan alat penyalin aman seperti [scp](#). Parameter `my-private-key` dalam perintah berikut adalah kunci privat yang Anda gunakan untuk terhubung ke instans Anda dengan SSH. Sebagai contoh:

```
you@your_computer:~ $ scp -i my-private-key.pem /
path/to/pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem ec2-
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
pk-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem 100% 717 0.7KB/s 00:00
```

3. Konfigurasi variabel lingkungan Anda untuk menggunakan AWS CLI. Untuk informasi selengkapnya, lihat [Membuat pasangan kunci](#).

- 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.

```
[ec2-user ~]$ aws ec2 create-volume --size 10 --region us-west-2 --
availability-zone us-west-2b
```

- b. Lampirkan volume ke instans yang didukung Amazon EBS menggunakan perintah [attach-volume](#).

```
[ec2-user ~]$ aws ec2 attach-volume --volume-id volume_id --instance-
id instance_id --device /dev/sdb --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 my-s3-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 volume_id --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 volume_id
```

- b. Periksa untuk memastikan kelengkapan snapshot Anda.

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

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

```
[ec2-user bundle]$ aws ec2 describe-images --region us-west-2 --image-id ami-id --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=snapshot_id} --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 volume_id
```

Referensi alat AMI

Anda dapat menggunakan perintah alat AMI untuk membuat dan mengelola AMI Linux yang didukung penyimpanan instans. Untuk menyiapkan alat, lihat [Mengatur alat AMI](#).

Untuk informasi tentang kunci akses Anda, lihat [Praktik Terbaik untuk akun AWS](#) di Panduan Referensi AWS Account Management .

Perintah

- [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 Linux yang didukung penyimpanan instans 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 menuju file kunci RSA yang dienkode PEM. 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 menuju sertifikat kunci publik Amazon EC2 X.509 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 [Mengatur alat AMI](#), sertifikat berada di `$EC2_AMIT00L_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-HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -c cert-
HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.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 Linux yang didukung oleh penyimpanan instans dengan mengompresi, mengenkripsi, dan menandatangani salinan volume perangkat root untuk instans tersebut.

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 AMI Linux yang didukung penyimpanan instans](#).

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 diencode PEM.

Wajib: Ya

jalur -k, --privatekey

Jalur menuju file kunci RSA yang diencode PEM milik 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 menuju sertifikat kunci publik Amazon EC2 X.509 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 [Mengatur alat AMI](#), sertifikat berada di \$EC2_AMIT00L_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 tidak ditentukan, Amazon EC2 akan memaketkan `/etc/fstab`.

Wajib: Tidak

`--generate-fstab`

Membuat paketan volume menggunakan `fstab` yang disediakan Amazon EC2.

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 klon. 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`

Kunci akses AWS rahasia.

Wajib: Ya

`token -t, --delegation-token`

Token delegasi untuk diteruskan ke AWS permintaan. Untuk informasi selengkapnya, lihat [Menggunakan Kredensial Keamanan Sementara](#).

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 awalnya 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 DOC-EXAMPLE-BUCKET1 -a your_access_key_id -
s your_secret_access_key
Deleting files:
DOC-EXAMPLE-BUCKET1/image.manifest.xml
DOC-EXAMPLE-BUCKET1/image.part.00
DOC-EXAMPLE-BUCKET1/image.part.01
DOC-EXAMPLE-BUCKET1/image.part.02
DOC-EXAMPLE-BUCKET1/image.part.03
DOC-EXAMPLE-BUCKET1/image.part.04
DOC-EXAMPLE-BUCKET1/image.part.05
DOC-EXAMPLE-BUCKET1/image.part.06
Continue? [y/n]
y
Deleted DOC-EXAMPLE-BUCKET1/image.manifest.xml
Deleted DOC-EXAMPLE-BUCKET1/image.part.00
Deleted DOC-EXAMPLE-BUCKET1/image.part.01
Deleted DOC-EXAMPLE-BUCKET1/image.part.02
```

```
Deleted DOC-EXAMPLE-BUCKET1/image.part.03
Deleted DOC-EXAMPLE-BUCKET1/image.part.04
Deleted DOC-EXAMPLE-BUCKET1/image.part.05
Deleted DOC-EXAMPLE-BUCKET1/image.part.06
ec2-delete-bundle complete.
```

ec2-download-bundle

Deskripsi

Mengunduh AMI Linux tertentu yang didukung penyimpanan instans 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

Kunci akses AWS 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 `DOC-EXAMPLE-BUCKET1`.

```
[ec2-user ~]$ mkdir bundled
[ec2-user ~]$ ec2-download-bundle -b DOC-EXAMPLE-BUCKET1/bundles/bundle_name
-m image.manifest.xml -a your_access_key_id -s your_secret_access_key -k pk-
HKZYKTAIG2ECMXIYIBH3HXV4ZBEXAMPLE.pem -d mybundle
Downloading manifest image.manifest.xml from DOC-EXAMPLE-BUCKET1 to mybundle/
image.manifest.xml ...
Downloading part image.part.00 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.00 ...
Downloaded image.part.00 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.01 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.01 ...
Downloaded image.part.01 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.02 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.02 ...
Downloaded image.part.02 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.03 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.03 ...
Downloaded image.part.03 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.04 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.04 ...
Downloaded image.part.04 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.05 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.05 ...
Downloaded image.part.05 from DOC-EXAMPLE-BUCKET1
Downloading part image.part.06 from DOC-EXAMPLE-BUCKET1/bundles/bundle_name to
mybundle/image.part.06 ...
Downloaded image.part.06 from DOC-EXAMPLE-BUCKET1
```

ec2-migrate-manifest

Deskripsi

Memodifikasi AMI Linux yang didukung penyimpanan instans (misalnya, sertifikat, kernel, dan disk RAM) agar 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 menuju file kunci RSA yang dienkode PEM milik 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

Kunci akses AWS 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 menuju sertifikat kunci publik Amazon EC2 X.509 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 [Mengatur alat 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 PV-GRUB dibandingkan 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 PV-GRUB dibandingkan 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-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CL0.pem --privatekey pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CL0.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**

Membuat ulang paketan dari AMI Linux yang didukung penyimpanan instans.

Sintaksis

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

Opsi

jalur `-k`, `--privatekey`

Jalur menuju file kunci RSA yang diencode PEM.

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 paketan untuk AMI Linux yang didukung oleh penyimpanan instans ke Amazon S3 dan menetapkan daftar kontrol akses (ACL) yang sesuai pada objek yang diunggah. Untuk informasi selengkapnya, lihat [Buat AMI Linux yang didukung penyimpanan instans](#).

Note

Untuk mengunggah objek ke bucket S3 untuk AMI Linux yang didukung penyimpanan instans 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 [Mengendalikan kepemilikan objek yang diunggah menggunakan Kepemilikan Objek S3](#).

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 rahasia Anda.

Wajib: Ya

token -t, --delegation-token

Token delegasi untuk diteruskan ke AWS permintaan. Untuk informasi selengkapnya, lihat [Menggunakan Kredensial Keamanan Sementara](#).

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 DOC-EXAMPLE-BUCKET1/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 DOC-EXAMPLE-BUCKET1 ...
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.

Buat AMI dengan Windows Sysprep

Alat Microsoft System Preparation (Sysprep) menyederhanakan proses duplikasi instalasi Windows kustomisasi. Anda dapat menggunakan Sysprep untuk membuat Amazon Machine Image (AMI) terstandarisasi. Anda kemudian dapat membuat instans Amazon EC2 baru untuk Windows dari gambar terstandarisasi ini.

Kami menyarankan Anda menggunakan [EC2 Image Builder](#) untuk mengotomatiskan pembuatan, pengelolaan, dan penyebaran gambar server yang disesuaikan, aman, up-to-date dan “emas” yang telah diinstal sebelumnya dan dikonfigurasi sebelumnya dengan perangkat lunak dan pengaturan.

Jika Anda menggunakan Windows Sysprep untuk membuat AMI standar, kami sarankan Anda menjalankan Sysprep dengan [EC2Launch v2](#) Jika Anda masih menggunakan agen EC2Config

(Windows Server 2012 R2 dan yang sebelumnya) atau EC2Launch (Windows Server 2016 dan 2019), lihat dokumentasi untuk menggunakan Sysprep dengan EC2Config dan EC2Launch di bawah ini.

Important

Jangan gunakan Sysprep untuk membuat cadangan instans. Sysprep menghapus informasi khusus sistem; menghapus informasi ini mungkin memiliki konsekuensi yang tidak diinginkan untuk pencadangan instans.

Untuk memecahkan masalah Sysprep, lihat [Memecahkan masalah Sysprep dengan instance Windows](#).

Daftar Isi

- [Sebelum Anda memulai](#)
- [Menggunakan Sysprep dengan EC2Launch v2](#)
- [Menggunakan Sysprep dengan EC2Launch](#)
- [Menggunakan Sysprep dengan EC2Config](#)

Sebelum Anda memulai

- Sebelum melakukan Sysprep, kami sarankan Anda menghapus semua akun pengguna lokal dan semua profil akun selain akun administrator tunggal yang akan menjalankan Sysprep. Jika Anda melakukan Sysprep dengan akun dan profil tambahan, dapat terjadi perilaku tak terduga, termasuk hilangnya data profil atau kegagalan untuk menyelesaikan Sysprep.
- Pelajari selengkapnya tentang [Sysprep](#) di Microsoft TechNet.
- Pelajari [peran server yang didukung untuk Sysprep](#).

Menggunakan Sysprep dengan EC2Launch v2

Bagian ini berisi rincian tentang berbagai fase eksekusi Sysprep dan tugas yang dilakukan oleh layanan EC2Launch v2 saat gambar disiapkan. Bagian ini juga mencakup langkah-langkah untuk membuat AMI standar menggunakan Sysprep dengan layanan EC2Launch v2.

Topik Sysprep dengan EC2Launch v2

- [Fase Sysprep](#)
- [Tindakan Sysprep](#)
- [Pasca Sysprep](#)
- [Menjalankan Sysprep dengan EC2Launch v2](#)

Fase Sysprep

Sysprep dijalankan melalui fase berikut:

- **Generalisasi:** Alat tersebut menghapus informasi dan konfigurasi spesifik gambar. Misalnya, Sysprep menghapus pengidentifikasi keamanan (SID), nama komputer, log peristiwa, dan driver khusus. Setelah fase ini selesai, sistem operasi (OS) siap untuk membuat AMI.

Note

Saat Anda menjalankan Sysprep dengan layanan EC2Launch v2, sistem mencegah driver dihapus karena pengaturan `PersistAllDeviceInstalls` diatur ke `true` secara default.

- **Spesialisasi:** Plug and Play memindai komputer dan menginstal driver untuk perangkat yang terdeteksi. Alat ini memunculkan persyaratan OS, seperti nama komputer dan SID. Anda juga dapat menjalankan perintah dalam fase ini.
- **Pengalaman Out-of-Box (OOBE):** Sistem menjalankan versi singkat Pengaturan Windows dan meminta Anda memasukkan informasi, seperti bahasa sistem, zona waktu, dan organisasi terdaftar. Saat menjalankan Sysprep dengan EC2Launch v2, file jawaban mengotomatisasi fase ini.

Tindakan Sysprep

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 tindakan keamanan yang diperlukan karena, selama sesi boot pertama setelah Sysprep berjalan, ada periode waktu singkat di mana RDP mengizinkan koneksi dan kata sandi Administrator kosong.

4. Layanan EC2Launch v2 memanggil 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 Amazon AMI memerlukan driver tertentu untuk dijalankan dan deteksi ulang driver tersebut akan memakan waktu.
 - `DoNotCleanUpNonPresentPerangkat`: Pengaturan ini menyimpan informasi Plug and Play untuk perangkat yang saat ini tidak ada.
- Sysprep akan 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 berdasarkan konfigurasi yang Anda tentukan dalam file jawaban `unattend.xml`.

- `CopyProfile`: 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 menggantikan profil default dengan profil administrator lokal yang ada. Semua akun yang masuk setelah menjalankan Sysprep akan menerima salinan profil dan isinya saat masuk pertama kali.

Jika Anda tidak memiliki kustomisasi profil pengguna tertentu yang ingin Anda tampilkan ke gambar baru tersebut, ubah pengaturan ini menjadi `False`. 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. Tindakan keamanan ini dirancang untuk mencegah instance agar tidak dapat diakses setelah Sysprep selesai jika Anda tidak mengonfigurasi 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_UNDEPLOYABLE

2. Sistem menjalankan EC2Launch v2.

Pasca Sysprep

Setelah 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 [Konfigurasi instans Windows menggunakan EC2Launch v2](#).

Menjalankan Sysprep dengan EC2Launch v2

Gunakan prosedur berikut ini untuk membuat AMI standar menggunakan Sysprep dan layanan EC2Launch v2.

1. Di konsol Amazon EC2, cari AMI yang ingin Anda duplikat.

2. Jalankan dan hubungkan ke instans Windows Anda.
3. Kustomisasikan.
4. Dari menu Start Windows, cari dan pilih pengaturan Amazon EC2Launch. Untuk informasi selengkapnya tentang opsi dan pengaturan di kotak dialog pengaturan EC2Launch Amazon, lihat [Pengaturan EC2Launch v2](#).
5. Pilih Matikan dengan Sysprep atau Matikan tanpa Sysprep.

Ketika Anda diminta untuk mengonfirmasi bahwa Anda ingin menjalankan Sysprep dan mematikan instans, klik Ya. EC2Launch v2 menjalankan 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 menginvokasi alat Sysprep dari baris perintah menggunakan perintah berikut:

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

Menggunakan Sysprep dengan EC2Launch

EC2Launch menawarkan file jawaban default dan file batch untuk Sysprep yang mengotomatiskan dan mengamankan proses persiapan gambar di 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 Sysprep untuk membuat cadangan instans. Sysprep menghapus informasi spesifik sistem. Jika Anda menghapus informasi ini, mungkin ada konsekuensi yang tidak diinginkan untuk pencadangan instans.

Sysprep dengan topik EC2Launch

- [File jawaban EC2Launch dan file batch untuk Sysprep](#)
- [Menjalankan Sysprep dengan EC2Launch](#)
- [Perbarui rute metadata/KMS untuk Server 2016 dan yang lebih baru saat meluncurkan AMI kustom](#)

File jawaban EC2Launch dan file batch untuk Sysprep

File jawaban EC2Launch dan file batch untuk Sysprep mencakup hal berikut:

`Unattend.xml`

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

`BeforeSysprep.cmd`

Kustomisasikan file batch ini untuk menjalankan perintah sebelum EC2Launch menjalankan Sysprep.

`SysprepSpecialize.cmd`

Kustomisasikan file batch ini untuk menjalankan perintah saat fase spesialisasi Sysprep.

Menjalankan Sysprep dengan EC2Launch

Pada instalasi penuh Windows Server 2016 dan yang lebih baru (dengan pengalaman desktop), Anda dapat menjalankan Sysprep dengan EC2Launch secara manual atau dengan menggunakan aplikasi Pengaturan Peluncuran EC2.

Untuk menjalankan 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 `LaunchSettings` aplikasi EC2. Aplikasi ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Settings`.

Ec2 Launch Settings

General

Set Computer Name

Set the computer name of the instance ip- <hex internal IP>. Disable this feature to persist your own computer name setting.

Set Wallpaper

Overlay instance information on the current wallpaper.

Extend Boot Volume

Extend OS partition to consume free space for boot volume.

Add DNS Suffix List

Add DNS suffix list to allow DNS resolution of servers running in EC2 without providing the fully qualified domain name.

Handle User Data

Execute user data provided at instance launch.
Note: This will be re-enabled when running shutdown with sysprep below.

Administrator Password

Random (Retrieve from console)

Specify (Temporarily store in config file)

Do Nothing (Customize Unattend.xml for sysprep)

These changes will take effect on next boot if Ec2Launch script is scheduled. By default, it is scheduled by shutdown options below.

Sysprep

Sysprep is a Microsoft tool that prepares an image for multiple launches.

Ec2Launch Script Location: **Found**

Run EC2Launch on every boot (instead of just the next boot).

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 di `LaunchConfig.json` sebagai teks polos dan dihapus setelah Sysprep mengatur 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 `unattend.xml` file. Jika Anda tidak menentukan kata sandi di `unattend.xml`, akun administrator akan dinonaktifkan.
7. Pilih Matikan dengan Sysprep.

Untuk menjalankan 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 membuat 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 EC2Launch 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.

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

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

- Jalankan EC2 LaunchSettings GUI (`C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe`) dan pilih opsi untuk mematikan dengan Sysprep.
- Jalankan EC2 LaunchSettings dan matikan tanpa 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 EC2 Peluncuran inisialisasi tugas secara manual 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 mengaktifasi Windows”](#).

Menggunakan Sysprep dengan EC2Config

Bagian ini berisi detail tentang berbagai fase pelaksanaan Sysprep dan tugas yang dilakukan oleh layanan EC2Config saat gambar disiapkan. Bagian ini juga mencakup langkah-langkah untuk membuat AMI standar dengan menggunakan Sysprep dengan layanan EC2Config.

Sysprep dengan topik EC2Config

- [Fase Sysprep](#)
- [Tindakan Sysprep](#)
- [Pasca Sysprep](#)
- [Menjalankan Sysprep dengan layanan EC2Config](#)

Fase Sysprep

Sysprep dijalankan melalui fase berikut:

- **Generalisasi:** Alat tersebut menghapus informasi dan konfigurasi spesifik gambar. Misalnya, Sysprep menghapus pengidentifikasi keamanan (SID), nama komputer, log peristiwa, dan driver khusus. Setelah fase ini selesai, sistem operasi (OS) siap untuk membuat AMI.

Note

Ketika Anda menjalankan Sysprep dengan layanan EC2config, sistem mencegah driver dihapus karena `PersistAllDeviceInstalls` pengaturan diatur ke `true` secara default.

- **Spesialisasi:** Plug and Play memindai komputer dan menginstal driver untuk perangkat yang terdeteksi. Alat ini memunculkan persyaratan OS, seperti nama komputer dan SID. Anda juga dapat menjalankan perintah dalam fase ini.
- **Pengalaman Out-of-Box (OOBE):** Sistem menjalankan versi singkat Pengaturan Windows dan meminta pengguna memasukkan informasi, seperti bahasa sistem, zona waktu, dan organisasi terdaftar. Saat menjalankan Sysprep dengan EC2Config, file jawaban mengotomatisasi fase ini.

Tindakan Sysprep

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 Sysprep secara otomatis. Anda tidak perlu mengubah nilai ini jika menjalankan 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 instance 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, mengenkripsi 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 untuk 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 tindakan keamanan yang diperlukan karena, selama sesi boot pertama setelah Sysprep berjalan, ada periode waktu singkat di mana RDP mengizinkan koneksi dan kata sandi Administrator kosong.

4. Layanan EC2Config memanggil 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 Amazon AMI memerlukan driver tertentu untuk dijalankan dan deteksi ulang driver tersebut akan memakan waktu.
 - `DoNotCleanUpNonPresentPerangkat`: Pengaturan ini menyimpan informasi Plug and Play untuk perangkat yang saat ini tidak ada.
- Sysprep akan 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`: 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` menggantikan profil default dengan profil administrator lokal yang ada. Semua akun yang masuk setelah menjalankan Sysprep akan menerima salinan profil dan isinya saat masuk pertama kali.

Apabila Anda tidak memiliki kustomisasi profil pengguna tertentu yang ingin Anda tampilkan ke gambar baru tersebut, ubah pengaturan ini menjadi `False`. 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-langkah keamanan ini dirancang untuk mencegah instans dapat diakses setelah Sysprep selesai jika Anda tidak mengaktifkan pengaturan ec2setpassword.

```
C:\Program Files\ Amazon\ Ec2ConfigService\ ScramblePassword .exe” -u Administrator
```

- Perintah sinkron dengan order 3: Sistem menjalankan perintah berikut:

```
C:\Program File\ Amazon\ Ec2\ ScriptConfigService\ Phase.cmd SysprepSpecialize
```

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

```
reg tambahkan “HKEY_LOCAL_MACHINE\ SYSTEM\ CurrentControl Set\ Control\ Terminal
Server” /v FdenytsConnections /t REG_DWORD /d 0/f
```

Fase OOB

1. Menggunakan file jawaban layanan EC2Config, sistem menentukan konfigurasi berikut ini:

- < InputLocale InputLocale >en-kami</ >
- < SystemLocale SystemLocale >en-kami</ >
- < UILanguage >en-US</UILanguage >
- < UserLocale UserLocale >en-kami</ >
- < HideEULAPage >true</HideEULAPage >
- < HideWireless SetupIn OOB >BENAR</ HideWireless SetupIn OOB >
- < NetworkLocation NetworkLocation >Lainnya</ >
- < ProtectYour PC > 3 </ PC > ProtectYour
- < BluetoothTaskbar IconEnabled BluetoothTaskbar IconEnabled >salah</ >
- < TimeZone TimeZone >UTC</ >
- < RegisteredOrganization RegisteredOrganization >Amazon.com</ >
- < RegisteredOwner RegisteredOwner >Amazon</ >

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:

```
EC2 ConfigMonitor State: 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 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.
 - Ec2 SetPassword
 - Nama Ec2 SetComputer
 - Ec2 InitializeDrives
 - Ec2 EventLog

- Ec2ConfigureRDP
 - Ec2OutputRDPcert
 - Surat Ec2 SetDrive
 - Ec2 WindowsActivate
 - Ec2 DynamicBoot VolumeSize
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 (Systems Manager)

Untuk informasi selengkapnya tentang plug-in Windows, lihat [Konfigurasi instance Windows menggunakan layanan EC2config \(legacy\)](#).

Menjalankan Sysprep dengan layanan EC2Config

Gunakan prosedur berikut ini untuk membuat AMI standar menggunakan 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 Start Windows, pilih All Programs, lalu pilih EC2 ConfigService Settings.
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.

- **Spesifikasikan:** Kata sandi disimpan dalam file jawaban Sysprep dalam bentuk tidak dienkripsi (teks polos). Saat Sysprep kemudian berjalan, ia mengatur 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.
- **Gunakan Yang Ada:** Kata sandi yang ada untuk akun Administrator tidak berubah saat Sysprep dijalankan atau EC2Config dimulai ulang. Penting untuk mengingat kata sandi ini karena Anda tidak dapat mengambilnya nanti.

8. Pilih OKE.

Ketika Anda diminta untuk mengonfirmasi bahwa Anda ingin menjalankan Sysprep dan mematikan instans, klik Ya. Anda akan melihat bahwa EC2Config menjalankan 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 menginvokasi alat 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 mengonfigurasi dan menjalankan Sysprep dari baris perintah, lihat Ec2ConfigService\Scripts\InstallUpdates.ps1.

Memodifikasi AMI

Anda dapat memodifikasi satu set atribut Amazon Machine Image (AMI) terbatas, seperti deskripsi AMI dan properti berbagi. Namun, konten AMI (data biner volume) tidak dapat dimodifikasi. Untuk memodifikasi konten AMI, Anda harus [membuat AMI baru](#).

Important

Anda tidak dapat memodifikasi konten (data biner volume) dari AMI yang didukung EBS karena snapshot yang mendukungnya tidak dapat diubah. Anda juga tidak dapat memodifikasi konten (data biner volume) dari AMI Linux yang didukung toko (didukung S3) instans karena konten ditandatangani, dan peluncuran instance akan gagal jika tanda tangan tidak cocok.

Untuk atribut AMI yang dapat dimodifikasi, lihat [ModifyImageAtribut](#) di Referensi API Amazon EC2.

Topik berikut memberikan petunjuk untuk menggunakan konsol Amazon EC2 dan AWS CLI memodifikasi atribut AMI:

- [Menjadikan AMI publik](#)
- [Membagikan AMI dengan organisasi atau unit organisasi tertentu](#)
- [Membagikan AMI kepada akun AWS tertentu](#)
- [Mengggunakan dukungan berbayar](#)
- [Konfigurasi AMI](#)

Menyalin AMI

Anda dapat menyalin Gambar Mesin Amazon (AMI) di dalam atau di seluruh AWS Wilayah. Anda dapat menyalin AMI yang didukung Amazon EBS dan AMI yang didukung toko instans. Anda dapat menyalin AMI yang didukung EBS dengan snapshot terenkripsi, dan juga mengubah status enkripsi selama proses penyalinan. Anda dapat menyalin AMI yang dibagikan dengan Anda.

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

Dengan AMI yang didukung EBS, 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

- [Pertimbangan](#)
- [Biaya](#)
- [Izin IAM](#)
- [Menyalin AMI](#)
- [Menghentikan operasi penyalinan AMI yang tertunda](#)
- [Penyalinan Lintas Wilayah](#)
- [Penyalinan lintas akun](#)
- [Enkripsi dan penyalinan](#)

Pertimbangan

- Izin untuk menyalin AMI — Anda dapat menggunakan kebijakan IAM untuk memberikan atau menolak izin pengguna untuk menyalin AMI. Izin tingkat sumber daya yang ditentukan untuk tindakan CopyImage hanya berlaku untuk AMI yang baru. Anda tidak dapat menentukan izin tingkat sumber daya untuk AMI sumber.
- Izin peluncuran dan izin bucket Amazon S3 AWS — tidak menyalin izin peluncuran atau izin bucket Amazon S3 dari sumber AMI 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 sumber AMI. 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 target dan snapshot backing nya.

Biaya

Tidak ada biaya untuk menyalin AMI. Namun, tarif penyimpanan standar dan transfer data berlaku. Jika Anda menyalin AMI yang didukung oleh EBS, Anda akan dikenai biaya untuk penyimpanan snapshot EBS tambahan.

Izin IAM

Untuk menyalin AMI yang didukung EBS atau instance store-backed AMI, Anda memerlukan izin IAM berikut:

- `ec2:CopyImage`— Untuk menyalin AMI. Untuk AMI yang didukung EBS, ia juga memberikan izin untuk menyalin snapshot dukungan AMI.
- `ec2:CreateTags`— Untuk menandai target AMI. Untuk AMI yang didukung EBS, ia juga memberikan izin untuk menandai snapshot dukungan AMI target.

Jika Anda menyalin AMI yang didukung penyimpanan instance, Anda memerlukan izin IAM tambahan berikut:

- `s3:CreateBucket`— Untuk membuat bucket S3 di Wilayah target untuk AMI baru
- `s3:GetBucketAcl`— Untuk membaca izin ACL untuk bucket sumber
- `s3:ListAllMyBuckets`— Untuk menemukan bucket S3 yang ada untuk AMI di Wilayah target
- `s3:GetObject`— Untuk membaca objek di ember sumber
- `s3:PutObject`— Untuk menulis objek di ember target
- `s3:PutObjectAcl`— Untuk menulis izin untuk objek baru di bucket target

Contoh kebijakan IAM untuk menyalin AMI yang didukung EBS dan menandai AMI target dan snapshot

Contoh kebijakan berikut memberi Anda izin untuk menyalin AMI yang didukung EBS dan menandai AMI target dan snapshot dukungannya.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "PermissionToCopyAllImages",
    "Effect": "Allow",
```

```

    "Action": [
      "ec2:CopyImage",
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*::image/*"
  }]
}

```

Contoh kebijakan IAM untuk menyalin AMI yang didukung EBS tetapi menolak menandai snapshot baru

`ec2:CopySnapshot` Izin secara otomatis diberikan ketika Anda mendapatkan `ec2:CopyImage` izin. Ini termasuk izin untuk menandai snapshot dukungan baru dari AMI target. Izin untuk menandai snapshot dukungan baru dapat ditolak secara eksplisit.

Contoh kebijakan berikut memberi Anda izin untuk menyalin AMI yang didukung EBS, tetapi menolak Anda untuk menandai snapshot dukungan baru dari AMI target.

```

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

```

Contoh kebijakan IAM untuk menyalin instance store-backed AMI dan menandai target AMI

Contoh kebijakan berikut memberi Anda izin untuk menyalin AMI instance store-backed AMI di bucket sumber yang ditentukan ke Wilayah tertentu, dan menandai AMI target.

```

{

```

```

"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:::ami-source-bucket/*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "s3:CreateBucket",
    "s3:GetBucketAcl",
    "s3:PutObjectAcl",
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::amis-for-account-in-region-hash"
  ]
}
]
}

```

Untuk mencari Amazon Resource Name (ARN) dari bucket sumber AMI, buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>, di panel navigasi pilih AMI, dan temukan nama bucket di kolom Sumber.

Note

s3:CreateBucketIzin hanya diperlukan saat pertama kali Anda menyalin instance store-backed AMI ke Wilayah individual. 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 menggunakan AWS Management Console, AWS Command Line Interface atau SDK, atau Amazon EC2 API, yang semuanya mendukung CopyImage tindakan tersebut.

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 Dinonaktifkan.
5. Pilih AMI yang akan disalin, lalu pilih Tindakan, Salin AMI.
6. Pada halaman Salin 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 [Penyalinan Lintas Wilayah](#).

- 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. (Hanya AMI yang didukung EBS) Enkripsi snapshot EBS dari salinan AMI: Pilih kotak centang ini untuk mengenkripsi snapshot target, atau untuk mengenkripsi ulang mereka menggunakan kunci yang berbeda. Jika enkripsi secara default diaktifkan, kotak centang Enkripsi snapshot EBS dari salinan AMI dipilih dan tidak dapat dihapus. Untuk informasi selengkapnya, lihat [Enkripsi dan penyalinan](#).
- f. (Hanya AMI yang didukung EBS) Kunci KMS: Kunci KMS yang digunakan untuk mengenkripsi snapshot target.
- g. Tag: Anda dapat menandai AMI baru dan snapshot baru dengan tag yang sama, atau Anda dapat menandai mereka dengan tag yang berbeda.
 - Untuk menandai AMI baru dan snapshot baru dengan tag yang sama, pilih Tag image dan snapshot bersama-sama. Tag yang sama diterapkan ke AMI baru dan setiap snapshot yang dibuat.
 - Untuk menandai AMI baru dan snapshot baru dengan tag yang berbeda, pilih Tag image 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.

- h. Saat Anda siap untuk menyalin AMI, pilih Salin AMI.

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

AWS CLI

Untuk menyalin AMI menggunakan AWS CLI

Anda dapat menyalin AMI menggunakan perintah [copy-image](#). Anda harus menentukan Wilayah sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `--source-region`. Anda dapat menentukan Wilayah tujuan menggunakan parameter `--region` atau

variabel lingkungan. Untuk informasi selengkapnya, lihat [Mengonfigurasi Antarmuka Baris AWS Perintah](#).

(Hanya AMI yang didukung EBS) Saat Anda mengenkripsi snapshot target selama penyalinan, Anda harus menentukan parameter tambahan ini: dan. `--encrypted --kms-key-id`

Untuk melihat contoh perintah, lihat [Contoh](#) di bawah [copy-image](#) di Referensi Perintah AWS CLI .

PowerShell

Untuk menyalin AMI menggunakan Alat untuk Windows PowerShell

Anda dapat menyalin AMI menggunakan [Copy-EC2Image](#) perintah. Anda harus menentukan Wilayah sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `-SourceRegion`. Anda dapat menentukan Wilayah tujuan menggunakan parameter `-Region` atau perintah `Set-AWSDefaultRegion`. Untuk informasi selengkapnya, lihat [Menentukan AWS Wilayah](#).

(Hanya AMI yang didukung EBS) Saat Anda mengenkripsi snapshot target selama penyalinan, Anda harus menentukan parameter tambahan ini: dan. `-Encrypted -KmsKeyId`

Menghentikan operasi penyalinan AMI yang tertunda

Anda dapat menghentikan salinan AMI yang tertunda menggunakan AWS Management Console atau baris perintah.

Console

Untuk menghentikan operasi penyalinan AMI menggunakan konsol

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 Actions, Deregister AMI.
5. Saat diminta konfirmasi, pilih Batalkan pendaftaran AMI.

Command line

Untuk menghentikan operasi penyalinan AMI menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

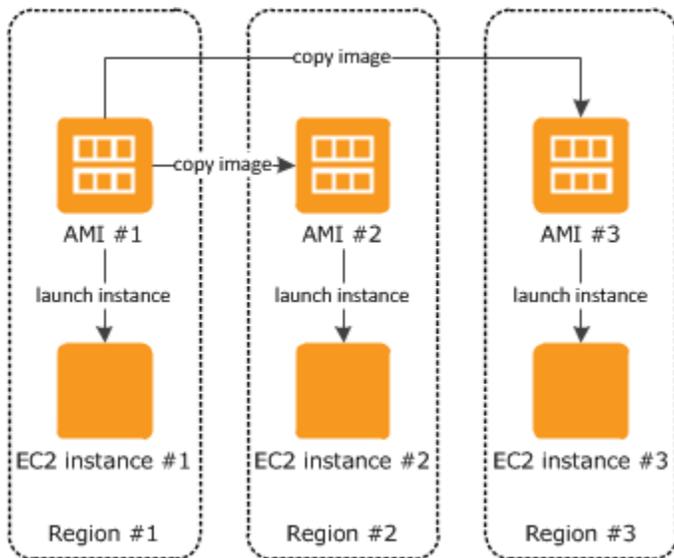
- [deregister-image](#) (AWS CLI)
- [Unregister-EC2Image](#) (AWS Tools for Windows PowerShell)

Penyalinan Lintas Wilayah

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 fitur khusus Wilayah, seperti jenis instans atau layanan lainnya AWS .
- 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 Anda pertama kali menyalin AMI yang didukung penyimpanan instans ke suatu Wilayah, kami membuat bucket Amazon S3 untuk AMI yang disalin ke Wilayah tersebut. Semua AMI yang didukung penyimpanan instans yang disalin ke Wilayah tersebut akan disimpan dalam bucket ini. Nama bucket memiliki format berikut: `ami-untuk-akun-di-wilayah-hash`. Sebagai contoh: `amis-for-123456789012-in-us-east-2-yhjmvp6`.

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 menjadi 100 salinan AMI secara bersamaan.
- Anda tidak dapat menyalin AMI paravirtual (PV) ke Wilayah yang tidak mendukung AMI PV. Untuk informasi selengkapnya, lihat [Tipe virtualisasi AMI](#).

Penyalinan lintas akun

Anda dapat berbagi AMI dengan AWS akun lain. Membagikan AMI tidak memengaruhi kepemilikan AMI. Akun pemilik dikenai biaya untuk penyimpanan di Wilayah. Untuk informasi selengkapnya, lihat [Membagikan AMI kepada akun AWS tertentu](#).

Jika Anda menyalin AMI yang telah dibagikan ke akun Anda, Anda adalah pemilik AMI target di akun Anda. Pemilik AMI sumber dikenai biaya transfer Amazon EBS atau Amazon S3 standar, dan Anda dikenai biaya penyimpanan AMI target di Wilayah tujuan.

Izin Sumber Daya

Untuk menyalin AMI yang dibagikan kepada Anda dari akun lain, pemilik AMI sumber harus memberi Anda izin baca untuk penyimpanan yang mendukung AMI tersebut. Penyimpanan dapat berupa snapshot EBS terkait (untuk AMI yang didukung Amazon EBS) atau bucket S3 terkait (untuk AMI yang didukung penyimpanan instans). Jika AMI bersama memiliki snapshot terenkripsi, pemilik harus membagikan kunci kepada Anda juga. Untuk informasi selengkapnya tentang pemberian izin sumber daya, untuk snapshot EBS, lihat Membagikan [snapshot Amazon EBS di Panduan Pengguna Amazon EBS](#), dan untuk bucket S3, lihat Manajemen identitas dan [akses di Amazon S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon](#).

Note

Untuk menyalin AMI dengan tag-nya, Anda harus memiliki izin peluncuran untuk AMI sumber.

Enkripsi dan penyalinan

Tabel berikut ini menunjukkan dukungan enkripsi untuk berbagai skenario penyalinan AMI. 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 terenkripsi-ke-tidak terenkripsi	Ya
2	Terenkripsi-ke-terenkripsi	Ya
3	Tidak terenkripsi-ke-terenkripsi	Ya
4	Terenkripsi-ke-tidak terenkripsi	Tidak

Note

Mengenkripsi selama tindakan CopyImage hanya berlaku pada AMI yang didukung Amazon EBS. Karena AMI yang didukung penyimpanan instans tidak bergantung pada snapshot, Anda tidak dapat menggunakan penyalinan untuk mengubah status enkripsinya.

Secara default (yaitu, tanpa menentukan parameter enkripsi), snapshot AMI cadangan disalin dengan status enkripsi asalnya. Menyalin AMI yang didukung oleh snapshot yang tidak dienkripsi menghasilkan snapshot target identik yang juga tidak dienkripsi. Jika AMI sumber didukung oleh snapshot terenkripsi, menyalinnya menghasilkan snapshot target identik yang dienkripsi oleh kunci yang sama. AWS KMS Menyalin AMI yang didukung oleh beberapa snapshot akan menyimpan, secara default, status enkripsi sumber di setiap snapshot target.

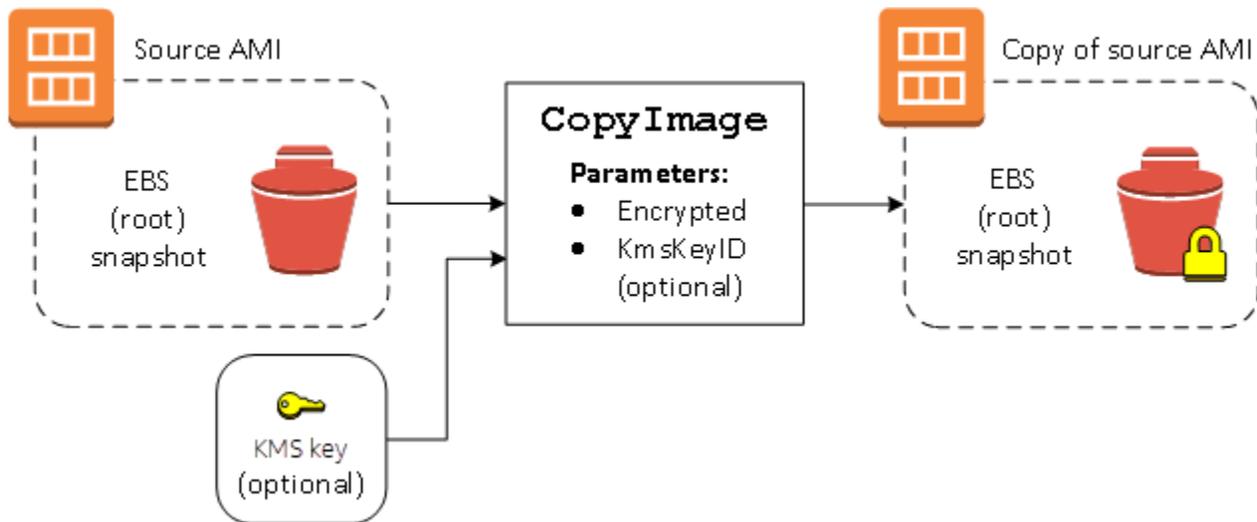
Jika Anda menentukan parameter enkripsi saat menyalin AMI, Anda dapat mengenkripsi atau mengenkripsi ulang snapshot cadangannya. Contoh berikut ini menunjukkan kasus non-default yang memasok parameter enkripsi ke tindakan CopyImage untuk mengubah status enkripsi AMI target.

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 mengatur Encrypted 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 [Menggunakan enkripsi dengan AMI yang didukung EBS](#).

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 partisi. AWS GovCloud (US) 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 *partisi* di halaman [Amazon Resource Names \(ARN\)](#) di Panduan Pengguna IAM.

⚠ Warning

Pastikan Anda mematuhi semua hukum dan persyaratan bisnis yang berlaku saat memindahkan data antar AWS partisi atau AWS Wilayah, termasuk, namun tidak terbatas pada, peraturan pemerintah dan persyaratan residensi data yang berlaku.

Topik

- [Kasus penggunaan](#)
- [Cara kerja API penyimpanan dan pemulihan AMI](#)
- [Batasan](#)
- [Biaya](#)
- [Mengamankan AMI Anda](#)
- [Izin untuk menyimpan dan memulihkan AMI menggunakan S3](#)
- [Bekerja dengan API penyimpanan dan AMI](#)
- [Menggunakan jalur file di S3](#)

Kasus penggunaan

Gunakan penyimpanan dan pemulihan API untuk melakukan hal berikut:

- [Salin AMI dari satu AWS partisi ke AWS partisi lain](#)
- [Buat salinan arsip AMI](#)

Salin AMI dari satu AWS partisi ke AWS partisi lain

Dengan menyimpan dan memulihkan AMI menggunakan bucket S3, Anda dapat menyalin AMI dari satu AWS partisi ke partisi lainnya, atau dari satu AWS Wilayah ke wilayah lainnya. Dalam contoh berikut, Anda menyalin AMI dari partisi komersial utama ke AWS GovCloud (US) partisi, khususnya dari us-east-2 Wilayah ke us-gov-east-1 Wilayah.

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. Untuk contoh perintah, lihat [Menyimpan AMI dalam bucket S3](#).

- Pantau kemajuan tugas penyimpanan dengan menggunakan `DescribeStoreImageTasks`. Objek akan terlihat dalam bucket S3 ketika tugas selesai. Untuk contoh perintah, lihat [Menggambarkan kemajuan tugas penyimpanan AMI](#).
- 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 kredensi 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 [unduh 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`. Untuk contoh perintah, lihat [Memulihkan AMI dari bucket S3](#).
 - 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. Untuk contoh perintah, lihat [Menyimpan AMI 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](#)

Cara kerja API 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](#)

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 AMI non-spesifik Wilayah, 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 600 GB data snapshot EBS per akun. Penciptaan tugas lebih lanjut akan ditolak sampai tugas yang sedang berlangsung kurang

dari batasan tersebut. Sebagai contoh, jika AMI dengan 100 GB data snapshot dan AMI lain dengan 200 GB data snapshot saat ini sedang disimpan, maka permintaan lain akan diterima, karena total yang sedang berlangsung adalah 300 GB, kurang dari batas. Tetapi jika sebuah AMI dengan 800 GB data snapshot saat ini sedang disimpan, maka tugas-tugas selanjutnya akan ditolak sampai tugas penyimpanan AMI 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 permintaan. [CreateStoreImageTask](#)

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 berlangsung dibatasi sampai 300 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.

Batasan

- Untuk menyimpan AMI, Anda Akun AWS harus memiliki AMI dan fotonya, atau AMI dan fotonya harus [dibagikan langsung dengan akun Anda](#). Anda tidak dapat menyimpan AMI jika AMI tersebut hanya [dibagikan secara publik](#).
- Hanya AMI yang didukung EBS yang dapat disimpan menggunakan API ini.
- AMI Paravirtual (PV) tidak didukung.
- Ukuran AMI (sebelum kompresi) yang dapat disimpan dibatasi hingga 5.000 GB.
- Kuota pada permintaan [simpan gambar](#): 600 GB penyimpanan (data snapshot) yang berlangsung.
- Kuota pada permintaan [pulihkan gambar](#): 300 GB pemulihan (data snapshot) yang 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 store dan restore API 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](#).

Mengamankan AMI Anda

Untuk menggunakan API penyimpanan dan pemulihan, bucket S3 dan AMI harus berada di Wilayah yang sama. 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 dianjurkan. Pastikan bahwa akses publik ke bucket S3 tidak diperbolehkan. Kami menyarankan untuk mengaktifkan [Enkripsi Sisi Server](#) untuk bucket S3 tempat Anda menyimpan AMI, meskipun tidak wajib.

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 kebijakan contoh 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.

```
{  
  "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": "*"
  }
]
```

Bekerja dengan API penyimpanan dan AMI

Topik

- [Menyimpan AMI dalam bucket S3](#)
- [Menggambarkan kemajuan tugas penyimpanan AMI](#)
- [Memulihkan AMI dari bucket S3](#)

Menyimpan AMI dalam bucket S3

Untuk menyimpan AMI (AWS CLI)

Gunakan perintah [create-store-image-task](#). Tentukan ID AMI dan nama bucket S3 untuk menyimpan AMI.

```
aws ec2 create-store-image-task \  
  --image-id ami-1234567890abcdef0 \  
  --bucket myamibucket
```

Output yang diharapkan

```
{  
  "ObjectKey": "ami-1234567890abcdef0.bin"  
}
```

Menggambarkan kemajuan tugas penyimpanan AMI

Untuk menggambarkan kemajuan tugas penyimpanan AMI (AWS CLI)

Gunakan perintah [describe-store-image-tasks](#).

```
aws ec2 describe-store-image-tasks
```

Output yang diharapkan

```
{  
  "AmiId": "ami-1234567890abcdef0",  
  "Bucket": "myamibucket",  
  "ProgressPercentage": 17,  
  "S3ObjectKey": "ami-1234567890abcdef0.bin",  
  "StoreTaskState": "InProgress",  
  "StoreTaskFailureReason": null,  
  "TaskStartTime": "2021-01-01T01:01:01.001Z"  
}
```

Memulihkan AMI dari bucket S3

Untuk memulihkan AMI (AWS CLI)

Gunakan perintah [create-restore-image-task](#). Menggunakan nilai untuk S3ObjectKey dan Bucket dari output `describe-store-image-tasks`, tentukan kunci objek AMI dan nama bucket S3 tempat AMI disalin. Tentukan juga nama untuk AMI yang dipulihkan. Nama harus unik untuk AMI di Wilayah untuk akun ini.

Note

AMI yang dipulihkan akan mendapat ID AMI baru.

```
aws ec2 create-restore-image-task \  
  --object-key ami-1234567890abcdef0.bin \  
  --bucket myamibucket \  
  --name "New AMI Name"
```

Output yang diharapkan

```
{  
  "ImageId": "ami-0eab20fe36f83e1a8"  
}
```

Menggunakan jalur file di S3

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.

Anda dapat menggunakan jalur file saat menggunakan AWS CLI dan SDK.

Contoh: Gunakan jalur file saat menyimpan dan memulihkan AMI (AWS CLI)

Contoh berikut ini pertama-tama menyimpan AMI di S3, dengan jalur file ditambahkan ke nama bucket. Contoh ini kemudian memulihkan AMI dari S3, dengan jalur file ditambahkan ke parameter kunci objek.

1. Menyimpan AMI. Untuk `--bucket`, tentukan jalur file setelah nama bucket, sebagai berikut:

```
aws ec2 create-store-image-task \  
  --image-id ami-1234567890abcdef0 \  
  --bucket myamibucket \  
  --object-key ami-1234567890abcdef0.bin
```

```
--bucket myamibucket/path1/path2
```

Output yang diharapkan

```
{  
  "ObjectKey": "path1/path2/ami-1234567890abcdef0.bin"  
}
```

2. Memulihkan AMI. Untuk `--object-key`, tentukan nilai dari output pada langkah sebelumnya, yang mencakup jalur file.

```
aws ec2 create-restore-image-task \  
  --object-key path1/path2/ami-1234567890abcdef0.bin \  
  --bucket myamibucket \  
  --name "New AMI Name"
```

Membuat usang sebuah AMI

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 tidak digunakan lagi tidak muncul di daftar AMI, mencegah pengguna baru menggunakan AMI out-of-date. 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 [DescribeImages](#) dalam panggilan 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 [DescribeImages](#) panggilan API.
- Untuk pengguna AMI, AMI usang tidak tersedia untuk dipilih melalui 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.

Anda juga dapat membuat kebijakan AMI yang didukung Amazon Data Lifecycle Manager untuk mengotomatiskan penghentian AMI yang didukung EBS. Untuk informasi selengkapnya, lihat [Mengotomatiskan siklus hidup AMI](#).

Note

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](#).

Topik

- [Biaya](#)
- [Batasan](#)
- [Membuat usang sebuah AMI](#)
- [Menggambarkan AMI yang diusangkan](#)
- [Membatalkan pengusangan 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.

Batasan

- Untuk mengusangkan AMI, Anda harus merupakan pemilik AMI tersebut.

Membuat usang sebuah AMI

Anda dapat mengusangkan AMI pada tanggal dan waktu tertentu. Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

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 usang.

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.

6. Pilih Simpan.

AWS CLI

Untuk mengusangkan AMI pada tanggal tertentu

Gunakan perintah [enable-image-deprecation](#). Tentukan ID AMI serta tanggal dan waktu untuk mengusangkan AMI. Jika Anda menentukan nilai untuk detik, Amazon EC2 membulatkan detik ke menit terdekat.

Batas atas untuk `deprecate-at` adalah 10 tahun dari sekarang, kecuali untuk AMI publik, di mana batas atas adalah 2 tahun dari tanggal pembuatan. Anda tidak dapat menentukan tanggal di masa lalu.

```
aws ec2 enable-image-deprecation \
```

```
--image-id ami-1234567890abcdef0 \  
--deprecate-at "2021-10-15T13:17:12.000Z"
```

Output yang diharapkan

```
{  
  "Return": "true"  
}
```

Periksa kapan AMI terakhir digunakan

`LastLaunchedTime` adalah stempel waktu yang menunjukkan kapan AMI Anda terakhir digunakan untuk meluncurkan instans. AMI yang belum digunakan baru-baru ini untuk meluncurkan instans mungkin merupakan kandidat yang baik untuk diusangkan atau [dibatalkan pendaftarannya](#).

Note

- Ketika AMI digunakan untuk meluncurkan instans, ada penundaan 24 jam sebelum penggunaan tersebut dilaporkan.
- `LastLaunchedTime` data tersedia mulai April 2017.

Console

Untuk melihat waktu peluncuran AMI terakhir

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 centang bidang Waktu peluncuran terakhir (jika Anda memilih kotak centang di sebelah AMI, ini terletak di tab Detail). Bidang menunjukkan tanggal dan waktu kapan AMI terakhir digunakan untuk meluncurkan instans.

AWS CLI

Untuk melihat waktu peluncuran AMI terakhir

Jalankan perintah `describe-image-attribute` dan tentukan `--attribute lastLaunchedTime`. Anda harus merupakan pemilik AMI untuk menjalankan perintah ini.

```
aws ec2 describe-image-attribute \  
  --image-id ami-1234567890example \  
  --attribute lastLaunchedTime
```

Contoh Output

```
{  
  "LastLaunchedTime": {  
    "Value": "2022-02-10T02:03:18Z"  
  },  
  "ImageId": "ami-1234567890example",  
}
```

Menggambarkan AMI yang diusangkan

Anda dapat melihat tanggal dan waktu pengusangan AMI, dan memfilter semua AMI berdasarkan tanggal pengusangan. Anda juga dapat menggunakan AWS CLI untuk mendeskripsikan semua AMI yang telah usang, di mana tanggal penghentian di masa lalu.

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. Periksa bidang Waktu pengusangan (jika Anda memilih kotak centang di sebelah AMI, ini 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 menggambarkan semua AMI menggunakan perintah [describe-images](#), hasilnya berbeda tergantung pada apakah Anda adalah pengguna AMI atau pemilik AMI.

- Jika Anda adalah pengguna AMI:

Secara default, ketika Anda menggambarkan semua AMI menggunakan perintah [describe-images](#), AMI usang yang tidak dimiliki oleh Anda, tetapi yang dibagi dengan Anda, tidak muncul dalam hasil. Ini karena default-nya adalah `--no-include-deprecated`. Untuk memasukkan AMI usang dalam hasil, Anda harus menentukan parameter `--include-deprecated`.

- Jika Anda adalah pemilik AMI:

Ketika Anda menggambarkan semua AMI menggunakan perintah [describe-images](#), semua AMI yang Anda miliki, termasuk AMI usang, muncul di hasil. Anda tidak perlu menentukan parameter `--include-deprecated`. Selain itu, Anda tidak dapat mengecualikan AMI usang yang Anda miliki dari hasil dengan menggunakan `--no-include-deprecated`.

Jika AMI diusangkan, bidang `DeprecationTime` akan muncul dalam hasil.

Note

AMI usang adalah AMI yang tanggal pengusangannya di masa lalu. Jika Anda telah menetapkan tanggal pengusangan ke tanggal di masa mendatang, maka AMI tersebut belum usang.

Untuk menyertakan semua AMI usang ketika menggambarkan semua AMI

Gunakan perintah [describe-images](#) dan tentukan parameter `--include-deprecated` untuk menyertakan semua AMI usang yang tidak dimiliki oleh Anda dalam hasil.

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --owners 123456example
```

```
--include-deprecated
```

Untuk melihat tanggal pengusangan sebuah AMI

Gunakan perintah [describe-images](#) dan tentukan ID AMI.

Perhatikan bahwa jika Anda menentukan `--no-include-deprecated` bersama dengan ID AMI, AMI usang akan muncul dalam hasil.

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --image-ids ami-1234567890EXAMPLE
```

Output yang diharapkan

Bidang `DeprecationTime` menampilkan tanggal AMI akan diusangkan. Jika AMI tidak diatur untuk diusangkan, maka bidang `DeprecationTime` tidak muncul di output.

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "PlatformDetails": "Red Hat Enterprise Linux",  
      "EnaSupport": true,  
      "Hypervisor": "xen",  
      "State": "available",  
      "SriovNetSupport": "simple",  
      "ImageId": "ami-1234567890EXAMPLE",  
      "DeprecationTime": "2021-05-10T13:17:12.000Z",  
      "UsageOperation": "RunInstances:0010",  
      "BlockDeviceMappings": [  
        {  
          "DeviceName": "/dev/sda1",  
          "Ebs": {  
            "SnapshotId": "snap-111222333444aaabb",  
            "DeleteOnTermination": true,  
            "VolumeType": "gp2",  
            "VolumeSize": 10,  
            "Encrypted": false  
          }  
        }  
      ],  
    }  
  ],  
}
```

```
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
  }
]
}
```

Membatalkan pengusangan AMI

Anda dapat membatalkan pengusangan sebuah AMI, yang menghapus tanggal dan waktu dari bidang Waktu pengusangan (konsol) atau bidang `DeprecationTime` dari output [describe-images](#) (AWS CLI). 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 Simpan.

AWS CLI

Untuk membatalkan pengusangan AMI

Gunakan perintah [disable-image-deprecation](#) dan tentukan ID AMI.

```
aws ec2 disable-image-deprecation \
  --image-id ami-1234567890abcdef0
```

Output yang diharapkan

```
{
  "Return": "true"
}
```

Menonaktifkan AMI

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.

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.

Anda dapat menonaktifkan AMI privat dan publik.

Topik

- [Biaya](#)
- [Prasyarat](#)
- [Izin IAM yang diperlukan](#)
- [Menonaktifkan AMI](#)
- [Menggambarkan AMI yang dinonaktifkan](#)
- [Aktifkan kembali AMI yang dinonaktifkan](#)

Biaya

Saat Anda menonaktifkan sebuah AMI, AMI tersebut tidak dihapus. Jika AMI adalah AMI yang didukung oleh EBS, Anda terus membayar 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 [Hapus sumber daya yang terkait dengan AMI yang didukung Amazon EBS-backed](#).

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 [disable-image](#) dan sebutkan ID AMI.

```
aws ec2 disable-image --image-id ami-1234567890abcdef0
```

Output yang diharapkan

```
{  
  "Return": "true"  
}
```

Menggambarkan AMI yang dinonaktifkan

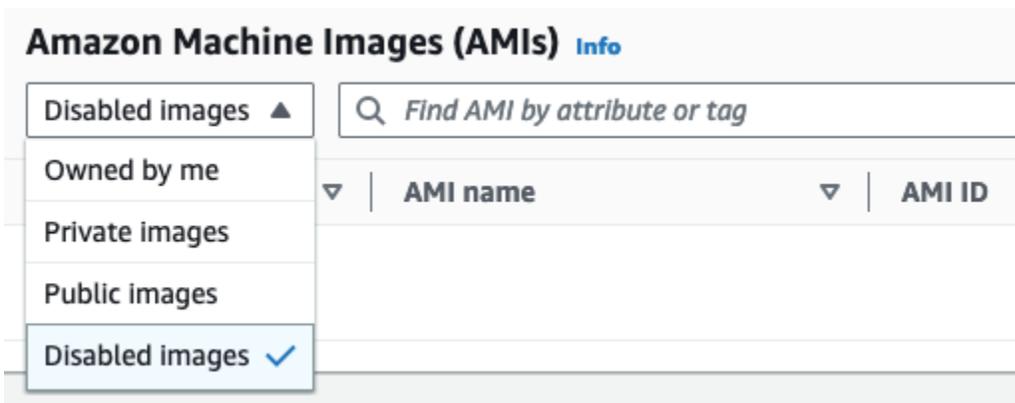
Anda dapat melihat AMI yang dinonaktifkan di konsol EC2 dan dengan menggunakan AWS CLI.

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 yang dinonaktifkan.



AWS CLI

Secara default, ketika Anda menggunakan perintah [describe-images](#) untuk menggambarkan semua AMI, AMI yang dinonaktifkan tidak muncul di hasil. Ini karena default-nya adalah `--no-include-disabled`. Untuk memasukkan AMI usang dalam hasil, Anda harus menentukan parameter `--include-disabled`.

Untuk menyertakan semua AMI yang dinonaktifkan ketika menggambarkan semua AMI

Gunakan perintah [describe-images](#) dan tentukan parameter `--include-disabled` untuk mengambil AMI yang dinonaktifkan selain semua AMI lainnya. Secara opsional, tentukan `--owners self` untuk hanya mengambil AMI yang Anda miliki.

```
aws ec2 describe-images \
```

```
--region us-east-1 \  
--owners self  
--include-disabled
```

Jika Anda menentukan ID AMI yang dinonaktifkan, tetapi tidak menentukan `--include-disabled`, AMI yang dinonaktifkan akan muncul dalam hasil.

```
aws ec2 describe-images \  
  --region us-east-1 \  
  --image-ids ami-1234567890EXAMPLE
```

Untuk mengambil hanya AMI yang dinonaktifkan

Tentukan `--filters Name=state,Values=disabled`. Anda juga harus menetapkan `--include-disabled`, jika tidak, Anda akan mendapatkan kesalahan.

```
aws ec2 describe-images \  
  --include-disabled \  
  --filters Name=state,Values=disabled
```

Contoh Output

Bidang `State` menampilkan status AMI. `disabled` menunjukkan bahwa AMI dinonaktifkan.

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "PlatformDetails": "Red Hat Enterprise Linux",  
      "EnaSupport": true,  
      "Hypervisor": "xen",  
      "State": "disabled",  
      "SriovNetSupport": "simple",  
      "ImageId": "ami-1234567890EXAMPLE",  
      "DeprecationTime": "2023-05-10T13:17:12.000Z",  
      "UsageOperation": "RunInstances:0010",  
      "BlockDeviceMappings": [  
        {  
          "DeviceName": "/dev/sda1",  
          "Ebs": {  
            "SnapshotId": "snap-111222333444aaabb",
```

```

        "DeleteOnTermination": true,
        "VolumeType": "gp2",
        "VolumeSize": 10,
        "Encrypted": false
      }
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": false,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
  }
]
}

```

Aktifkan kembali AMI yang dinonaktifkan

Anda dapat mengaktifkan 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 yang 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 [enable-image](#) dan sebutkan ID AMI.

```
aws ec2 enable-image --image-id ami-1234567890abcdef0
```

Output yang diharapkan

```
{  
  "Return": "true"  
}
```

Mengarsipkan snapshot AMI

Anda dapat mengarsipkan snapshot yang terkait dengan AMI yang didukung EBS yang dinonaktifkan. Ini dapat membantu Anda mengurangi biaya penyimpanan yang terkait dengan AMI yang jarang digunakan yang perlu dipertahankan untuk waktu yang lama. Untuk informasi selengkapnya, lihat [Mengarsipkan snapshot Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Untuk mengarsipkan snapshot yang terkait dengan AMI

1. [Nonaktifkan AMI](#).
2. [Arsipkan snapshot](#).

Anda tidak dapat menggunakan AMI saat dinonaktifkan dan snapshot terkait diarsipkan.

Untuk memulihkan AMI yang dinonaktifkan dengan snapshot yang diarsipkan untuk digunakan

1. [Kembalikan snapshot yang diarsipkan](#) terkait dengan AMI.
2. [Aktifkan AMI](#).

Deregister (hapus) AMI

Saat Anda membatalkan pendaftaran AMI, Amazon EC2 menghapusnya secara permanen. Setelah dideregistrasi, Anda tidak dapat menggunakan AMI 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 secara tidak sengaja membatalkan pendaftaran

AMI yang didukung EBS, Anda dapat menggunakan [Recycle Bin](#) untuk memulihkannya hanya jika Anda memulihkannya dalam jangka waktu yang diizinkan sebelum dihapus secara permanen.

Membatalkan pendaftaran AMI tidak berpengaruh pada instans apa pun yang diluncurkan dari AMI. Anda dapat terus menggunakan contoh ini. Deregistering 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 instance apa pun dan menghapus snapshot apa pun yang tidak Anda perlukan. Untuk informasi selengkapnya, lihat [Hindari biaya dari sumber daya yang tidak terpakai](#).

Daftar Isi

- [Pertimbangan](#)
- [Membatalkan pendaftaran AMI](#)
- [Periksa kapan AMI terakhir digunakan](#)
- [Lindungi AMI dari deregistrasi](#)
- [Hindari biaya dari sumber daya yang tidak terpakai](#)

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

Gunakan salah satu metode berikut untuk membatalkan pendaftaran AMI yang didukung EBS atau AMI yang didukung oleh instance store-backed.

Tip

Untuk menghindari biaya yang tidak perlu, Anda harus menghapus sumber daya apa pun yang tidak Anda butuhkan. Misalnya, untuk AMI yang didukung EBS, jika Anda tidak memerlukan snapshot yang terkait dengan AMI yang dideregistrasi, Anda harus

menghapusnya. Untuk informasi selengkapnya, lihat [Hindari biaya dari sumber daya yang tidak terpakai](#).

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 Dimiliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar yang dinonaktifkan untuk mencantumkan AMI Anda yang dinonaktifkan.
4. Pilih AMI untuk membatalkan pendaftaran.
5. Pilih Tindakan, Batalkan pendaftaran AMI.
6. Ketika Anda diminta untuk konfirmasi, pilih Deregister 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](#) dan tentukan ID AMI untuk membatalkan pendaftaran.

```
aws ec2 deregister-image --image-id ami-0123456789example
```

Powershell

Untuk membatalkan pendaftaran AMI

Gunakan [Unregister-EC2Image](#)cmdlet dan tentukan ID AMI untuk membatalkan pendaftaran.

```
Unregister-EC2Image -ImageId ami-0123456789example
```

Periksa kapan AMI terakhir digunakan

LastLaunchedTime adalah stempel waktu yang menunjukkan kapan AMI Anda terakhir digunakan untuk meluncurkan instans. AMI yang belum digunakan baru-baru ini untuk meluncurkan instans mungkin merupakan kandidat yang baik untuk dibatalkan pendaftarannya atau [diusangkan](#).

Note

- Ketika AMI digunakan untuk meluncurkan instans, ada penundaan 24 jam sebelum penggunaan tersebut dilaporkan.
- Data lastLaunchedTime 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 AMI, lalu centang bidang Waktu peluncuran terakhir (jika Anda memilih kotak centang di sebelah AMI, ini terletak di tab Detail). Bidang menunjukkan tanggal dan waktu kapan AMI terakhir digunakan untuk meluncurkan instans.

AWS CLI

Anda dapat menggunakan perintah [describe-images](#) atau [describe-image-attribute](#) untuk melihat waktu peluncuran terakhir AMI.

Untuk melihat waktu peluncuran terakhir AMI dengan menggunakan gambar-deskripsikan

Gunakan perintah [describe-images](#) dan tentukan ID AMI.

```
aws ec2 describe-images --image-id ami-0123456789example
```

Contoh Output

Note

LastLaunchedTimeBidang hanya muncul di output untuk AMI yang Anda miliki.

```
{
  "Images": [
    {
      ...
      "LastLaunchedTime": {
        "Value": "2024-04-02T02:03:18Z"
      },
      ...
    }
  ]
}
```

Untuk melihat waktu peluncuran AMI terakhir

Gunakan [perintah describe-image-attribute](#) dan tentukan. `--attribute lastLaunchedTime` Anda harus menjadi pemilik AMI untuk menjalankan perintah ini.

```
aws ec2 describe-image-attribute \
  --image-id ami-0123456789example \
  --attribute lastLaunchedTime
```

Contoh Output

```
{
  "ImageId": "ami-1234567890example",
  "LastLaunchedTime": {
    "Value": "2022-02-10T02:03:18Z"
  }
}
```

Lindungi AMI dari deregistrasi

Anda dapat mengaktifkan perlindungan deregistrasi pada AMI untuk mencegah penghapusan yang tidak disengaja atau berbahaya. Saat Anda mengaktifkan perlindungan deregistrasi, AMI tidak dapat

dideregistrasi oleh pengguna mana pun, terlepas dari izin IAM mereka. Jika Anda ingin membatalkan pendaftaran AMI, Anda harus mematikan perlindungan deregistrasi terlebih dahulu.

Saat Anda 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 mematakannya. Selama periode cooldown ini, AMI tidak dapat dideregistrasi. Ketika periode cooldown berakhir, AMI dapat dideregistrasi.

Perlindungan deregistrasi dimatikan secara default pada semua AMI yang ada dan yang baru.

Aktifkan perlindungan deregistrasi

Gunakan salah satu metode berikut untuk mengaktifkan perlindungan deregistrasi pada AMI. Untuk melakukan ini, Anda harus menjadi pemilik AMI.

Console

Untuk mengaktifkan perlindungan deregistrasi pada AMI

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar yang dinonaktifkan untuk mencantumkan AMI Anda yang dinonaktifkan.
4. Pilih AMI tempat Anda ingin mengaktifkan perlindungan deregistrasi, lalu pilih Tindakan, Kelola perlindungan deregistrasi 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 dideregistrasi selama 24 jam saat perlindungan deregistrasi dimatikan.
 - Aktifkan tanpa cooldown — Tanpa periode cooldown, AMI dapat segera dideregistrasi saat perlindungan deregistrasi dimatikan.
6. Pilih Simpan.

AWS CLI

Untuk mengaktifkan perlindungan deregistrasi pada AMI

Gunakan perintah [enable-image-deregistration-protection](#) dan tentukan ID AMI. Untuk memasukkan periode cooldown 24 jam opsional, sertakan `--with-cooldown` set ke `true`. Untuk mengecualikan periode cooldown, hilangkan parameter. `--with-cooldown`

```
aws ec2 enable-image-deregistration-protection \  
  --image-id ami-0123456789example \  
  --with-cooldown true
```

Matikan perlindungan deregistrasi

Gunakan salah satu metode berikut untuk mematikan perlindungan deregistrasi pada AMI. Untuk melakukan ini, Anda harus menjadi pemilik AMI.

Note

Jika Anda memilih untuk menyertakan periode cooldown 24 jam ketika Anda mengaktifkan perlindungan deregistrasi untuk AMI, maka, ketika Anda mematikan perlindungan deregistrasi, Anda tidak akan segera dapat membatalkan pendaftaran AMI. Periode cooldown adalah periode waktu 24 jam di mana perlindungan deregistrasi tetap berlaku bahkan setelah Anda memamatkannya. Selama periode cooldown ini, AMI tidak dapat dideregistrasi. Setelah periode cooldown berakhir, AMI dapat dideregistrasi.

Console

Untuk mematikan perlindungan deregistrasi pada AMI

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar yang dinonaktifkan untuk mencantumkan AMI Anda yang dinonaktifkan.
4. Pilih AMI untuk mematikan perlindungan deregistrasi, lalu pilih Tindakan, Kelola perlindungan deregistrasi AMI.
5. Di kotak dialog Kelola perlindungan deregistrasi AMI, pilih Nonaktifkan.
6. Pilih Simpan.

AWS CLI

Untuk mematikan perlindungan deregistrasi pada AMI

Gunakan perintah [disable-image-deregistration-protection](#) dan tentukan ID AMI.

```
aws ec2 disable-image-deregistration-protection --image-id ami-0123456789example
```

Hindari biaya dari sumber daya yang tidak terpakai

Saat Anda membatalkan pendaftaran AMI, Anda tidak menghapus sumber daya yang terkait dengan AMI. Sumber daya ini termasuk snapshot untuk AMI yang didukung EBS dan file di Amazon S3 misalnya AMI yang didukung toko. 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 instance yang berjalan. Untuk informasi selengkapnya, lihat [Bagaimana Anda dikenai biaya](#).

Untuk menghindari timbulnya jenis biaya yang tidak perlu ini, kami sarankan untuk menghapus sumber daya apa pun yang tidak Anda perlukan.

Untuk menentukan apakah AMI Anda didukung EBS atau didukung oleh instans toko, lihat. [Tentukan tipe perangkat root AMI Anda](#)

Hapus sumber daya yang terkait dengan AMI yang didukung Amazon EBS-backed

Gunakan salah satu metode berikut untuk menghapus sumber daya yang terkait dengan AMI yang didukung EBS.

Console

Untuk menghapus sumber daya yang terkait dengan AMI yang didukung EBS

1. [Deregister AMI](#).

Perhatikan ID AMI—Ini dapat membantu Anda menemukan snapshot yang akan dihapus pada langkah berikutnya.

2. [Hapus snapshot](#) yang tidak Anda butuhkan.

ID AMI terkait ditampilkan di kolom Deskripsi di layar Snapshots.

3. [Hentikan contoh](#) yang tidak Anda butuhkan.

AWS CLI

Untuk menghapus sumber daya yang terkait dengan AMI yang didukung EBS

1. [Deregister AMI dengan menggunakan perintah deregister-image.](#)

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. Hapus snapshot yang tidak Anda butuhkan dengan menggunakan perintah [delete-snapshot](#).

```
aws ec2 delete-snapshot --snapshot-id snap-0123456789example
```

3. Hentikan instance yang tidak Anda butuhkan dengan menggunakan perintah [terminate-instance](#).

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

PowerShell

Untuk menghapus sumber daya yang terkait dengan AMI yang didukung EBS

1. Batalkan pendaftaran AMI dengan menggunakan cmdlet. [Unregister-EC2Image](#)

```
Unregister-EC2Image -ImageId ami-0123456789example
```

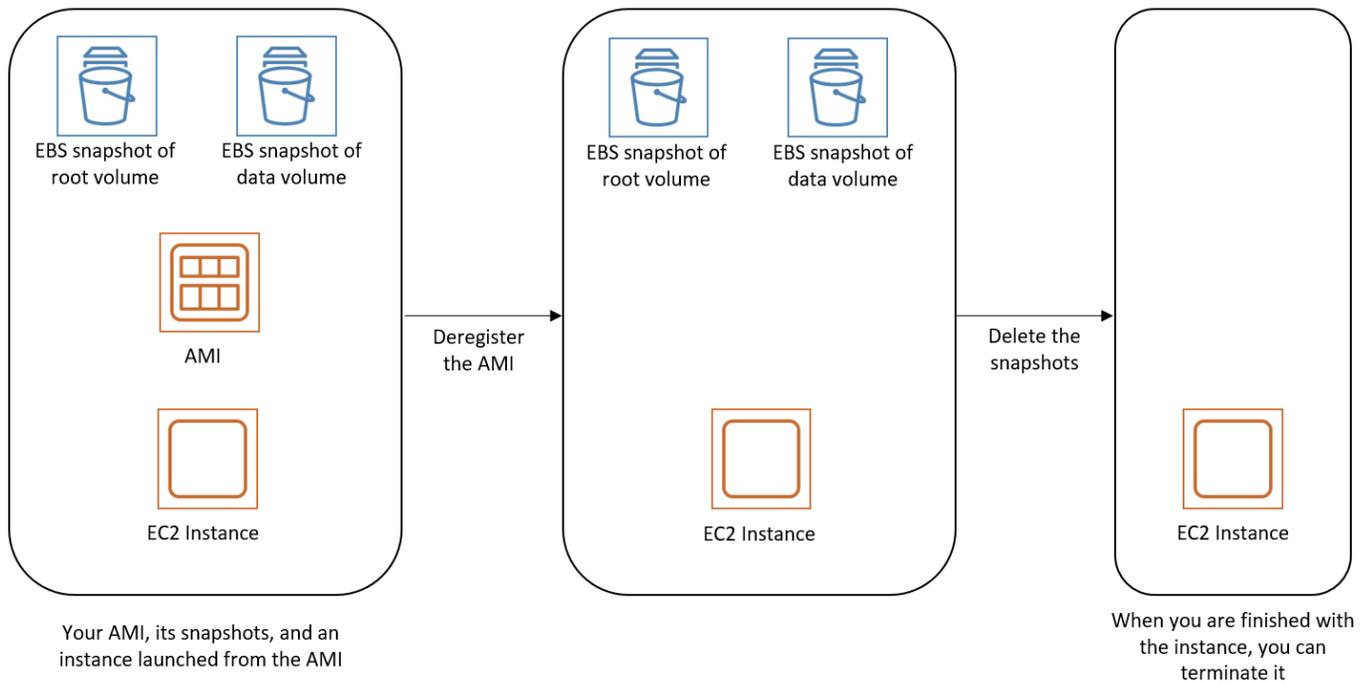
2. Hapus snapshot yang tidak Anda butuhkan dengan menggunakan [Remove-EC2Snapshot](#) cmdlet.

```
Remove-EC2Snapshot -SnapshotId snap-0123456789example
```

3. Hentikan instance yang tidak Anda butuhkan dengan menggunakan cmdlet. [Remove-EC2Instance](#)

```
Remove-EC2Instance -InstanceId i-0123456789example
```

Diagram berikut mengilustrasikan alur bagi Anda untuk menghapus sumber daya yang terkait dengan AMI yang didukung EBS.



Hapus sumber daya yang terkait dengan AMI yang didukung oleh instance store-backed

Gunakan metode berikut untuk menghapus sumber daya yang terkait dengan instance store-backed AMI Anda.

Untuk menghapus sumber daya yang terkait dengan AMI yang didukung oleh instance store-backed

1. [Deregister AMI dengan menggunakan perintah deregister-image.](#)

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. Hapus bundel di Amazon S3 dengan menggunakan perintah ([ec2-delete-bundle](#)AMI tools).

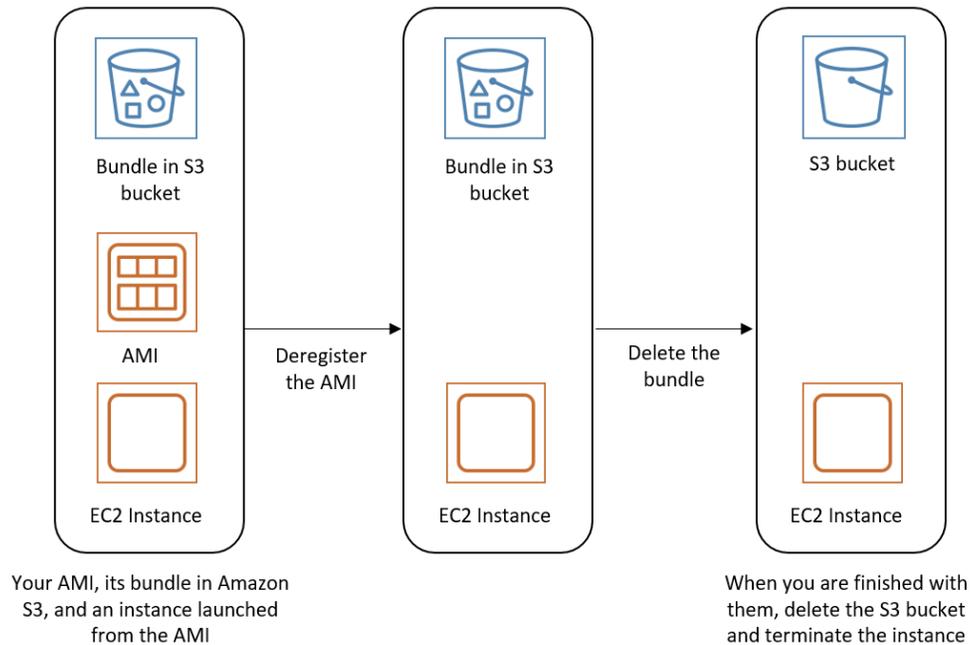
```
ec2-delete-bundle -b myawsbucket/myami -a your_access_key_id -s your_secret_access_key -p image
```

3. Hentikan instance yang tidak Anda butuhkan dengan menggunakan perintah [terminate-instance](#).

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

4. Jika Anda sudah selesai dengan bucket Amazon S3 tempat Anda mengunggah bundel, Anda dapat menghapus bucket. Untuk menghapus bucket Amazon S3, buka konsol Amazon S3, pilih bucket, pilih Tindakan, lalu pilih Hapus.

Diagram berikut mengilustrasikan alur bagi Anda untuk menghapus sumber daya yang terkait dengan instance store-backed AMI Anda.



Otomatisasi siklus hidup AMI yang didukung EBS

Anda dapat menggunakan Amazon Data Lifecycle Manager untuk melakukan otomatisasi pembuatan, penyimpanan, penyalinan, pengusangan, dan pembatalan pendaftaran AMI yang didukung Amazon EBS dan snapshot pendukungnya. Untuk informasi selengkapnya, lihat [Amazon Data Lifecycle Manager](#).

Menggunakan enkripsi dengan AMI yang didukung EBS

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 jika 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 baik oleh default Anda AWS KMS key, atau ke kunci terkelola pelanggan yang Anda tentukan. Anda harus selalu memiliki izin untuk menggunakan kunci KMS yang dipilih.

AMI dengan snapshot terenkripsi dapat dibagikan di seluruh akun. AWS Untuk informasi selengkapnya, lihat [AMI bersama](#).

Topik enkripsi dengan AMI yang didukung EBS

- [Skenario peluncuran instans](#)
- [Skenario penyalinan gambar](#)

Skenario peluncuran instans

Instans Amazon EC2 diluncurkan dari AMI menggunakan RunInstances tindakan dengan parameter yang disediakan melalui pemetaan perangkat blok, baik melalui AWS Management Console atau langsung menggunakan Amazon EC2 API atau CLI. Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#). Untuk contoh mengontrol pemetaan blok perangkat dari AWS CLI, lihat [Meluncurkan, Mendaftar, dan Mengakhiri Instans 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.

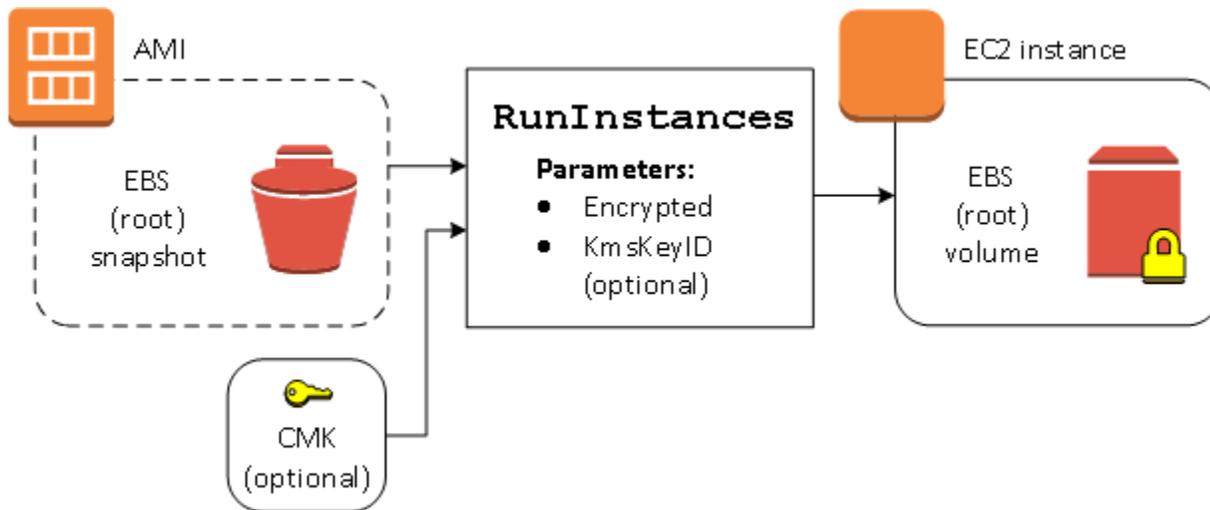
Mengirim `KmsKeyId` tanpa mengatur parameter `Encrypted` akan mengakibatkan kesalahan.

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 Anda](#).

Mengenkripsi volume saat peluncuran

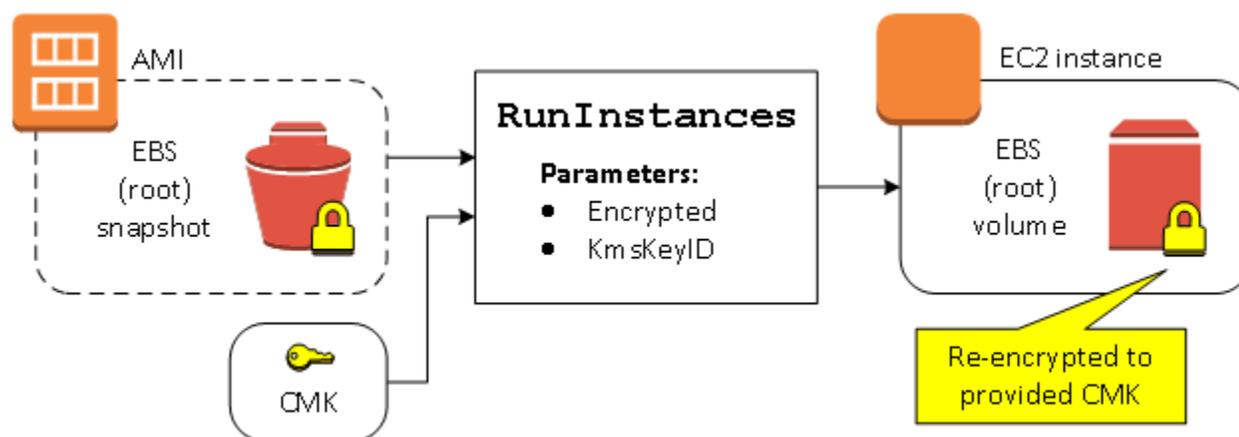
Dalam contoh ini, AMI yang didukung oleh snapshot tidak terenkripsi digunakan untuk meluncurkan instans EC2 dengan volume EBS terenkripsi.



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`.

Menkripsi ulang volume saat peluncuran

Dalam contoh ini, AMI yang didukung oleh snapshot terenkripsi digunakan untuk meluncurkan instans EC2 dengan volume EBS yang dienkripsi oleh kunci KMS baru.

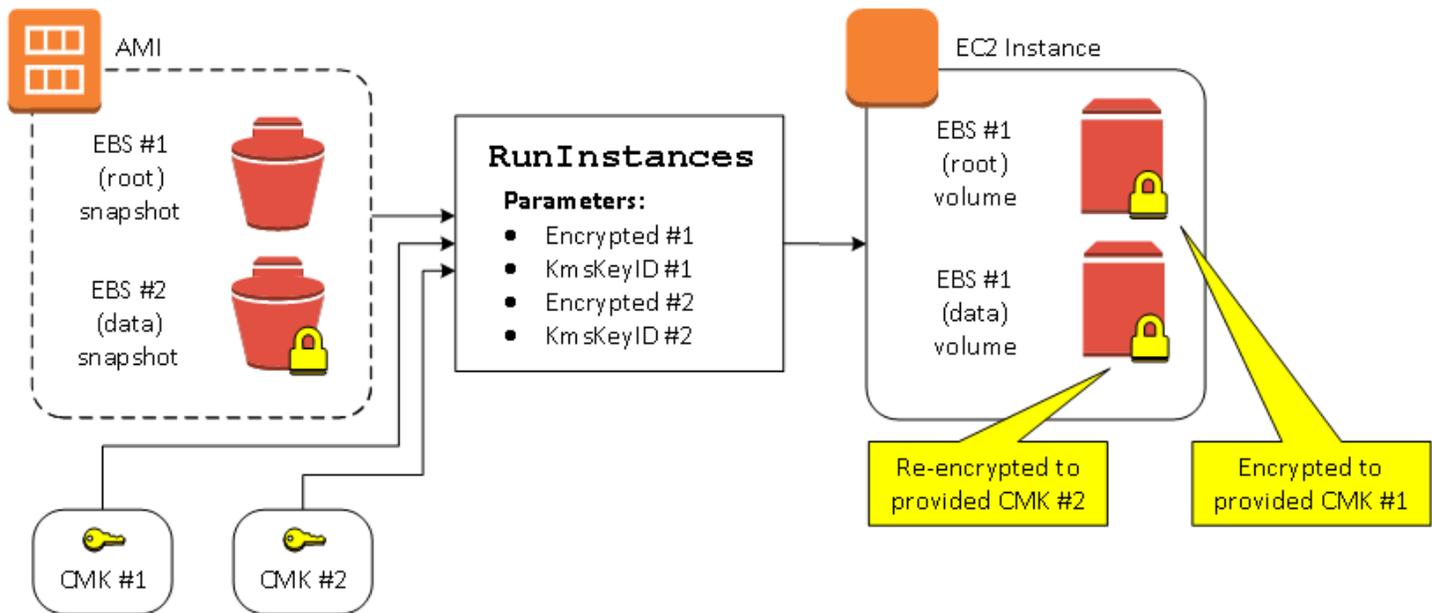


Jika Anda memiliki AMI dan tidak menyediakan parameter enkripsi, instans yang dihasilkan memiliki volume yang dienkripsi oleh kunci KMS yang sama dengan snapshot. Jika AMI adalah AMI bersama,

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.

Skenario penyalinan gambar

AMI Amazon EC2 disalin menggunakan tindakan CopyImage, baik melalui AWS Management Console 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 akun Anda. AWS

Perilaku default ini dapat ditimpa dengan menyediakan parameter enkripsi. Parameter yang tersedia adalah `Encrypted` dan `KmsKeyId`. Jika hanya menetapkan parameter `Encrypted`, hal berikut terjadi:

Perilaku copy-image dengan **Encrypted** diatur, tetapi tidak ada **KmsKeyId** yang 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 akun Anda AWS . (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:

Perilaku copy-image dengan **Encrypted** dan **KmsKeyId** diatur

- 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.

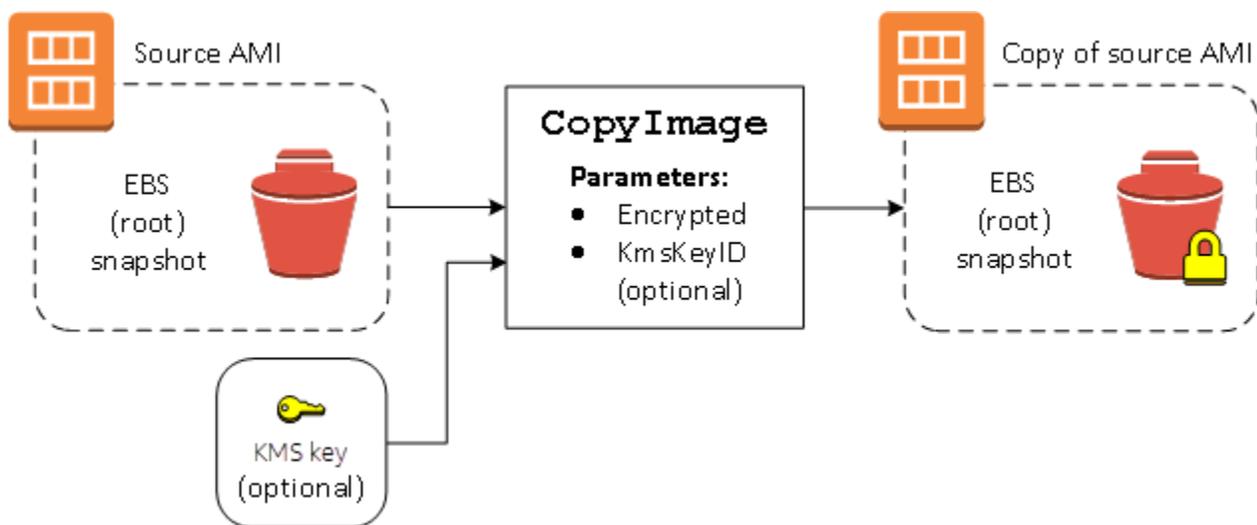
Untuk petunjuk detail menggunakan konsol, lihat [Menyalin AMI](#).

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 mengatur Encrypted 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.

Pantau peristiwa AMI menggunakan Amazon EventBridge

Ketika status Amazon Machine Image (AMI) berubah, Amazon EC2 menghasilkan peristiwa yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon Events). CloudWatch 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 peristiwa saat AMI memasuki salah satu status berikut:

- available
- failed
- deregistered
- disabled

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		
CreateRes toreImageTask	Ya	Ya		
DeregisterImage			Ya	
DisableImage				Ya
EnableImage	Ya			
RegisterImage	Ya	Ya		

Peristiwa dihasilkan atas dasar upaya terbaik.

Topik

- [Peristiwa AMI](#)
- [Buat EventBridge aturan Amazon](#)

Peristiwa AMI

Ada empat peristiwa EC2 AMI State Change:

- [available](#)
- [failed](#)
- [deregistered](#)
- [disabled](#)

Acara dikirim ke bus EventBridge acara default dalam format JSON.

Bidang berikut dalam peristiwa dapat digunakan 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-0123456789example", "State": "available", }
```

Memberikan informasi berikut ini:

- ID AMI – Jika Anda ingin melacak AMI tertentu.
- Status AMI (available, failed, deregistered, atau disabled).

available

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-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "available",
    "ErrorMessage": ""
  }
}
```

failed

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-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "failed",
  }
}
```

```
    "ErrorMessage": "Description of failure"
  }
}
```

deregistered

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-d8b8441bcd0",
  "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-0123456789example"],
  "detail": {
    "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
    "ImageId": "ami-0123456789example",
    "State": "deregistered",
    "ErrorMessage": ""
  }
}
```

disabled

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-d8b8441bcd0",
```

```

"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-0123456789example"],
"detail": {
  "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
  "ImageId": "ami-0123456789example",
  "State": "disabled",
  "ErrorMessage": ""
}
}

```

Buat EventBridge aturan Amazon

Anda dapat membuat EventBridge [aturan](#) Amazon yang menentukan tindakan yang akan diambil saat EventBridge menerima [peristiwa](#) yang cocok dengan [pola acara](#) dalam aturan. Saat acara cocok, EventBridge kirimkan acara ke [target](#) yang ditentukan dan memicu tindakan yang ditentukan dalam aturan.

Pola peristiwa memiliki struktur yang sama dengan peristiwa yang dicocokkan. Pola peristiwa bisa cocok atau tidak dengan peristiwa.

Saat membuat aturan untuk peristiwa perubahan status AMI, Anda dapat menyertakan bidang berikut dalam pola peristiwa:

```
"source": "aws.ec2"
```

Mengidentifikasi bahwa peristiwa tersebut dari Amazon EC2.

```
"detail-type": "EC2 AMI State Change"
```

Mengidentifikasi nama peristiwa.

```
"detail": { "ImageId": "ami-0123456789example", "State": "available", }
```

Memberikan informasi berikut ini:

- ID AMI – Jika Anda ingin melacak AMI tertentu.
- Status AMI (available, failed, deregistered, atau disabled).

Contoh: Buat EventBridge aturan untuk mengirim notifikasi

Contoh berikut membuat EventBridge aturan untuk mengirim email, pesan teks, atau pemberitahuan push seluler ketika AMI apa pun berada dalam `available` status setelah `CreateImage` operasi berhasil diselesaikan.

Sebelum membuat EventBridge aturan, Anda harus membuat topik Amazon SNS untuk email, pesan teks, atau notifikasi push seluler.

Untuk membuat EventBridge aturan untuk mengirim pemberitahuan ketika AMI dibuat dan di **available** negara bagian

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 sesuai dengan peristiwa EC2 AMI State Change apa pun yang dihasilkan saat AMI memasuki status `available`:

```
{
  "source": ["aws.ec2"],
  "detail-type": ["EC2 AMI State Change"],
  "detail": {"State": ["available"]}
}
```

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 EC2.
 - D. Untuk Tipe peristiwa, pilih Perubahan Status AMI EC2.
 - E. Untuk menyesuaikan templat, pilih Edit pola dan buat perubahan Anda agar sesuai dengan contoh pola peristiwa.
 - ii. 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 topik yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat [Menggunakan Amazon SNS untuk pesan application-to-person \(A2P\) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).
 - 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 topik berikut di Panduan EventBridge Pengguna Amazon:

- [EventBridge Acara Amazon](#)
- [Pola EventBridge acara Amazon](#)
- [EventBridge Aturan Amazon](#)

Untuk tutorial tentang cara membuat fungsi Lambda dan EventBridge aturan yang menjalankan fungsi Lambda, lihat [Tutorial: Mencatat status instans Amazon EC2 menggunakan dalam Panduan Pengembang](#). EventBridge AWS Lambda

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 garis bawah AWS tagihan 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. Sebagai contoh, RunInstances:0010. Operasi penggunaan [sesuai dengan kolom Lineltem/Operation pada Laporan AWS Biaya dan Penggunaan \(CUR\) Anda dan di API Daftar Harga.AWS](#)

Anda dapat melihat bidang ini di halaman Instans atau AMI di konsol Amazon EC2, atau dalam respons yang ditampilkan oleh [Get-EC2Image](#) gambar-gambar atau [perintah](#).

Contoh data: operasi penggunaan berdasarkan platform

Tabel berikut mencantumkan beberapa detail platform dan nilai operasi penggunaan yang dapat ditampilkan di halaman Instans atau AMI di konsol Amazon EC2, atau dalam respons yang ditampilkan oleh [Get-EC2Image](#) gambar-gambar atau [perintah](#).

Detail platform	Operasi penggunaan ²
Linux/UNIX	RunInstances
Red Hat BYOL Linux	RunInstances:00g0 ³
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

Detail platform	Operasi penggunaan ²
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
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 menunjukkan 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 [lineitem/Operation](#) ditampilkan `RunInstances:0010:SV006`, itu berarti Amazon EC2 menjalankan Red Hat Enterprise Linux Spot Instance-hour di US East (Virginia N.) di Zona 6.

³ Ini muncul seperti RunInstances (Linux/UNIX) dalam laporan penggunaan Anda.

Mencari detail penagihan dan penggunaan AMI

Di konsol Amazon EC2, Anda dapat melihat informasi penagihan AMI dari halaman AMI atau dari Instans. Anda juga dapat menemukan informasi penagihan menggunakan AWS CLI atau layanan metadata instans.

Bidang berikut dapat membantu Anda memverifikasi biaya AMI pada tagihan Anda:

- Detail platform
- Operasi penggunaan
- ID AMI

Mencari informasi penagihan AMI (konsol)

Ikuti langkah-langkah ini untuk melihat informasi penagihan AMI di konsol Amazon EC2:

Cari informasi penagihan AMI dari halaman AMI

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih AMI, lalu pilih AMI.
3. Di tab Detail, periksa nilai untuk Detail platform dan Operasi penggunaan.

Cari informasi penagihan AMI dari halaman Instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans, lalu pilih instans.
3. Pada tab Detail (atau tab Deskripsi jika Anda menggunakan konsol versi sebelumnya), periksa nilai untuk Detail platform dan Operasi penggunaan.

Mencari informasi penagihan AMI (AWS CLI)

Untuk menemukan informasi penagihan AMI menggunakan AWS CLI, Anda perlu mengetahui ID AMI. Jika Anda tidak tahu ID AMI, Anda bisa mendapatkannya dari instans menggunakan perintah [describe-instances](#).

Untuk mencari ID AMI

Jika Anda tidak tahu ID instans, Anda bisa mendapatkan ID AMI untuk instans menggunakan perintah [describe-instances](#).

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

Dalam output, ID AMI ditentukan dalam bidang ImageId.

```
... "Instances": [  
  {  
    "AmiLaunchIndex": 0,  
    "ImageId": "ami-0123456789EXAMPLE",  
    "InstanceId": "i-123456789abcde123",  
    ...  
  }  
]
```

Untuk mencari informasi penagihan AMI

Jika Anda mengetahui ID AMI, Anda dapat menggunakan perintah [describe-images](#) untuk mendapatkan detail platform AMI dan operasi penggunaan.

```
$ aws ec2 describe-images --image-ids ami-0123456789EXAMPLE
```

Contoh output berikut menunjukkan bidang PlatformDetails dan UsageOperation. Dalam contoh ini, platform ami-0123456789EXAMPLE adalah Red Hat Enterprise Linux dan operasi penggunaan serta kode penagihannya adalah RunInstances:0010.

```
{  
  "Images": [  
    {  
      "VirtualizationType": "hvm",  
      "Description": "Provided by Red Hat, Inc.",  
      "Hypervisor": "xen",  
      ...  
    }  
  ]  
}
```

```

    "EnaSupport": true,
    "SriovNetSupport": "simple",
    "ImageId": "ami-0123456789EXAMPLE",
    "State": "available",
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/sda1",
        "Ebs": {
          "SnapshotId": "snap-111222333444aaabb",
          "DeleteOnTermination": true,
          "VolumeType": "gp2",
          "VolumeSize": 10,
          "Encrypted": false
        }
      }
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "PlatformDetails": "Red Hat Enterprise Linux",
    "UsageOperation": "RunInstances:0010",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
  }
]
}

```

Memverifikasi biaya AMI pada tagihan Anda

Untuk memastikan bahwa Anda tidak menimbulkan biaya yang tidak direncanakan, Anda dapat memverifikasi bahwa informasi penagihan untuk instans dalam Laporan AWS Biaya dan Penggunaan (CUR) 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

Kuota berikut ini berlaku untuk membuat dan berbagi AMI. Kuota berlaku per Wilayah AWS.

Nama kuota	Deskripsi	Kuota default per Wilayah
AMI	Jumlah maksimum AMI publik dan privat yang diizinkan per Wilayah. Ini termasuk AMI yang tersedia, tertunda, dan dinonaktifkan, dan AMI di Keranjang Sampah.	50.000
AMI Publik	Jumlah maksimum AMI publik, termasuk AMI publik di Keranjang Sampah, yang diperbolehkan per Wilayah.	5
Berbagi AMI	Jumlah maksimum entitas (organisasi, unit organisasi (OU), dan akun) yang dapat dibagikan AMI di suatu Wilayah. Perhatikan bahwa jika Anda berbagi AMI dengan organisasi atau OU, jumlah akun dalam organisasi atau OU tidak dihitung dalam kuota.	1.000

Jika Anda melebihi kuota dan ingin membuat atau berbagi lebih banyak AMI, Anda dapat melakukan hal berikut:

- Jika Anda melebihi total kuota AMI atau AMI publik, pertimbangkan untuk membatalkan pendaftaran gambar yang tidak digunakan.
- Jika Anda melebihi kuota AMI publik Anda, pertimbangkan untuk menjadikan satu atau lebih AMI publik menjadi privat.
- Jika Anda melebihi kuota berbagi AMI, pertimbangkan untuk membagikan AMI Anda dengan organisasi atau OU, bukan akun terpisah.
- Meminta peningkatan kuota untuk AMI.

Meminta peningkatan kuota untuk AMI

Jika Anda membutuhkan lebih dari kuota default untuk AMI, Anda dapat meminta peningkatan kuota.

Untuk meminta peningkatan kuota AMI

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 ketikkan nama layanan di kotak pencarian.
4. Pilih kuota AMI untuk meminta kenaikan. Kuota AMI yang dapat Anda pilih adalah:
 - AMI
 - AMI Publik
 - 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

Sebelum Anda meluncurkan lingkungan produksi, Anda perlu menjawab pertanyaan berikut.

T. Tipe instans apa yang paling sesuai dengan kebutuhan saya?

Amazon EC2 menyediakan berbagai tipe instans untuk memungkinkan Anda memilih CPU, memori, penyimpanan, dan kapasitas jaringan yang Anda perlukan untuk menjalankan aplikasi. Untuk informasi selengkapnya, lihat [Jenis Instans Amazon EC2](#).

T. Opsi pembelian apa yang paling sesuai dengan kebutuhan saya?

Amazon EC2 mendukung Instans Sesuai Permintaan (default), Instans Spot, dan Instans Terpesan. Untuk informasi selengkapnya, lihat [Opsi pembelian instans](#).

T. Jenis volume root mana yang memenuhi kebutuhan saya?

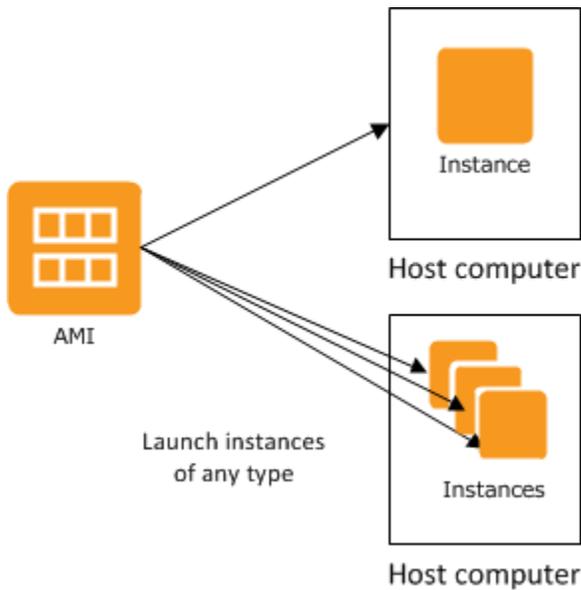
Setiap instans didukung oleh Amazon EBS atau didukung oleh penyimpanan instans. Pilih AMI berdasarkan tipe volume root yang Anda butuhkan. Untuk informasi selengkapnya, lihat [Penyimpanan untuk perangkat root](#).

T. Dapatkah saya mengelola armada instans EC2 dan mesin di lingkungan hibrida saya dari jarak jauh?

AWS Systems Manager memungkinkan Anda mengelola konfigurasi instans Amazon EC2 dari jarak jauh dan aman, serta instans lokal serta mesin virtual (VM) di lingkungan hybrid, termasuk VM dari penyedia cloud lainnya. Untuk informasi selengkapnya, lihat [Panduan Pengguna.AWS Systems Manager](#)

Instans dan AMI

Amazon Machine Image (AMI) adalah templat yang berisi konfigurasi perangkat lunak (misalnya, sistem operasi, server aplikasi, dan aplikasi). Dari AMI, luncurkan sebuah instans, yang merupakan salinan AMI yang dijalankan sebagai server virtual di cloud. Anda dapat meluncurkan beberapa instans AMI, seperti yang ditunjukkan pada gambar berikut.



Instans Anda terus berjalan hingga Anda menghentikan, hibernasi, atau mengakhirinya, atau hingga instans gagal. Jika sebuah instans gagal, Anda dapat meluncurkan instans baru dari AMI.

Instans

Sebuah instans adalah server virtual di cloud. Konfigurasinya saat peluncuran adalah salinan AMI yang Anda tentukan saat Anda meluncurkan instans.

Anda dapat meluncurkan berbagai tipe instans dari satu AMI. Tipe instans pada dasarnya menentukan perangkat keras komputer host yang digunakan untuk instans Anda. Setiap tipe instans menawarkan kemampuan komputasi dan memori yang berbeda. Pilih tipe instans berdasarkan jumlah memori dan daya komputasi yang Anda perlukan untuk aplikasi atau perangkat lunak yang Anda rencanakan untuk dijalankan pada instans. Untuk spesifikasi tipe instans terperinci, lihat [Spesifikasi](#) di Panduan Jenis Instans Amazon EC2. Untuk informasi harga, lihat [Harga Sesuai Permintaan Amazon EC2](#).

Setelah Anda meluncurkannya, instans akan tampak seperti host tradisional, dan Anda dapat berinteraksi dengannya seperti yang Anda lakukan di komputer mana pun. Anda memiliki kendali penuh atas instans Anda; Anda bisa memakai sudo untuk menjalankan perintah yang membutuhkan hak akses root.

AWS Akun Anda memiliki batasan jumlah instance yang dapat Anda jalankan. Untuk informasi selengkapnya tentang batas ini, dan cara meminta peningkatan, lihat [Berapa banyak instans yang dapat saya jalankan di Amazon EC2](#) di FAQ Umum Amazon EC2.

Penyimpanan untuk instans Anda

Perangkat root untuk instans Anda berisi gambar yang digunakan untuk boot instans tersebut. Perangkat root bisa berupa volume Amazon Elastic Block Store (Amazon EBS) atau volume penyimpanan instans. Untuk informasi selengkapnya, lihat [Volume root instans Amazon EC2](#).

Instans Anda mungkin menyertakan volume penyimpanan lokal, yang dikenal sebagai volume penyimpanan instans, yang dapat Anda konfigurasi pada waktu peluncuran dengan pemetaan perangkat blok. Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#). Setelah volume ini ditambahkan dan dipetakan pada instans Anda, volume tersebut tersedia untuk Anda pasang dan gunakan. Jika instans Anda gagal, atau jika instans Anda dihentikan atau diakhiri, data pada volume ini akan hilang; oleh karena itu, volume ini paling baik digunakan untuk data sementara. Untuk menjaga keamanan data penting, Anda harus menggunakan strategi replikasi di banyak instans, atau menyimpan data persisten Anda dalam volume Amazon S3 atau Amazon EBS. Untuk informasi selengkapnya, lihat [Opsi penyimpanan untuk instans Amazon EC2 Anda](#).

Praktik terbaik keamanan

- Gunakan AWS Identity and Access Management (IAM) untuk mengontrol akses ke AWS sumber daya Anda, termasuk instans Anda. Untuk informasi selengkapnya, lihat [Manajemen identitas dan akses untuk Amazon EC2](#).
- Batasi akses dengan hanya mengizinkan host atau jaringan tepercaya untuk mengakses port pada instans Anda. Misalnya, Anda dapat membatasi akses SSH dengan membatasi lalu lintas masuk pada port 22. Untuk informasi selengkapnya, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).
- Tinjau aturan di grup keamanan Anda secara teratur, dan pastikan bahwa Anda menerapkan prinsip hak akses paling rendah—hanya membuka izin yang Anda perlukan. Anda juga dapat membuat grup keamanan berbeda untuk menangani instans yang memiliki persyaratan keamanan berbeda. Pertimbangkan untuk membuat grup keamanan bastion yang memungkinkan login eksternal, dan pertahankan sisa instans Anda dalam grup yang tidak mengizinkan login eksternal.
- Nonaktifkan login berbasis kata sandi untuk instans yang diluncurkan dari AMI Anda. Kata sandi dapat ditemukan atau dipecahkan, dan merupakan risiko keamanan. Untuk informasi selengkapnya, lihat [Nonaktifkan login jarak jauh berbasis kata sandi untuk pengguna root](#). Untuk informasi selengkapnya tentang membagikan AMI dengan aman, lihat [AMI bersama](#).

Menghentikan dan mengakhiri instans

Anda dapat menghentikan atau mengakhiri instans yang sedang berjalan kapan saja.

Menghentikan instans

Saat sebuah instans dihentikan, instans tersebut melakukan pematian normal, dan kemudian bertransisi ke status `stopped`. Semua volume Amazon EBS-nya tetap terpasang, dan Anda dapat memulai instans lagi di lain waktu.

Anda tidak akan dikenai biaya untuk penggunaan instans tambahan saat instans dalam keadaan berhenti. Anda dikenakan biaya untuk setiap transisi dari status berhenti ke status berjalan. Jika jenis instans berubah saat instance dihentikan, Anda akan dikenakan tarif untuk jenis instans baru setelah instance dimulai. Anda juga dikenakan biaya untuk penyimpanan Amazon EBS terkait untuk instans Anda, termasuk volume perangkat root.

Saat instans dalam keadaan berhenti, Anda dapat memasang atau melepaskan volume Amazon EBS. Anda juga dapat membuat AMI dari instans, dan Anda dapat mengubah kernel, disk RAM, dan tipe instans.

Akhiri instans

Saat sebuah instans diakhiri, instans tersebut melakukan pematian normal. Volume perangkat root dihapus secara default, tetapi setiap volume Amazon EBS yang terpasang dipertahankan secara default, ditentukan oleh masing-masing pengaturan atribut `deleteOnTermination` volume. Instans itu sendiri juga dihapus, dan Anda tidak dapat memulai lagi di lain waktu.

Untuk mencegah penghentian yang tidak disengaja, Anda dapat menonaktifkan pengakhiran instans. Jika Anda melakukannya, pastikan bahwa atribut `disableApiTermination` diatur ke `true` untuk instans. Untuk mengontrol perilaku pematian instans, seperti `shutdown -h` di Linux atau `shutdown` di Windows, atur atribut instans `instanceInitiatedShutdownBehavior` ke `stop` atau `terminate` seperti yang diinginkan. Instans dengan volume Amazon EBS untuk perangkat root default ke `stop` dan instans dengan perangkat root penyimpanan instans selalu diakhiri sebagai akibat dari pematian instans.

Untuk informasi selengkapnya, lihat [Siklus hidup instans](#).

Note

Beberapa AWS sumber daya, seperti volume Amazon EBS dan alamat IP Elastis, dikenakan biaya terlepas dari status instans. Untuk informasi selengkapnya, lihat [Menghindari Biaya](#)

[Tidak Terduga](#) dalam AWS Billing Panduan Pengguna . Untuk informasi selengkapnya tentang biaya Amazon EBS, lihat [Harga Amazon EBS](#).

AMI

Amazon Web Services (AWS) menerbitkan Amazon Machine Images (AMI) yang berisi konfigurasi perangkat lunak umum untuk penggunaan umum. Selain itu, anggota komunitas AWS pengembang telah menerbitkan AMI kustom mereka sendiri. Anda juga dapat membuat AMI kustom Anda sendiri; melakukannya memungkinkan Anda untuk dengan cepat dan mudah memulai instance baru yang memiliki semua yang Anda butuhkan. Misalnya, jika aplikasi Anda adalah situs web atau layanan web, AMI Anda dapat menyertakan server web, konten statis terkait, dan kode untuk laman dinamis. Akibatnya, setelah Anda meluncurkan sebuah instans dari AMI ini, server web Anda dimulai, dan aplikasi Anda siap untuk menerima permintaan.

Semua AMI dikategorikan sebagai didukung oleh by Amazon EBS, yang berarti bahwa perangkat root untuk instans yang diluncurkan dari AMI adalah volume Amazon EBS, atau didukung oleh penyimpanan instans, yang berarti bahwa perangkat root untuk instans yang diluncurkan dari AMI adalah volume penyimpanan instans yang dibuat dari templat yang disimpan di Amazon S3.

Deskripsi AMI menunjukkan tipe perangkat root (baik ebs atau instance store). Hal ini penting karena ada perbedaan signifikan tentang apa yang dapat Anda lakukan dengan setiap tipe AMI. Untuk informasi selengkapnya tentang metrik ini, lihat [Penyimpanan untuk perangkat root](#).

Anda dapat membatalkan pendaftaran AMI setelah selesai digunakan. Setelah Anda membatalkan pendaftaran AMI, maka Anda tidak dapat menggunakan AMI tersebut untuk meluncurkan instans baru. Instans yang ada yang telah diluncurkan dari AMI tersebut tidak terpengaruh. Oleh karena itu, jika Anda juga selesai menggunakan instans yang diluncurkan dari AMI ini, Anda harus mengakhirinya.

Jenis 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.

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, tipe instans dengan performa I/O yang tinggi memiliki alokasi sumber daya bersama yang lebih besar. Mengalokasikan bagian sumber daya bersama yang lebih besar juga mengurangi varians performa I/O. Untuk sebagian besar aplikasi, performa I/O moderat sudah lebih dari cukup. Namun, untuk aplikasi yang membutuhkan performa I/O yang lebih besar atau lebih konsisten, pertimbangkan tipe instans dengan performa I/O yang lebih tinggi.

Daftar Isi

- [Tipe instans yang tersedia](#)
- [Spesifikasi perangkat keras](#)
- [Tipe virtualisasi AMI](#)
- [Menemukan tipe instans Amazon EC2](#)
- [Mendapatkan rekomendasi untuk tipe instans](#)
- [Ubah tipe instans](#)
- [Instans performa yang dapat melonjak](#)
- [Akselerasi kinerja dengan instans GPU](#)

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 jenis instans menyertakan satu atau beberapa ukuran instans, memungkinkan Anda untuk menskalakan sumber daya sesuai dengan persyaratan beban kerja target Anda. Untuk informasi selengkapnya tentang fitur dan kasus penggunaan, lihat [Detail Jenis Instans Amazon EC2](#).

Konvensi penamaan tipe instans

Nama didasarkan pada keluarga instance, 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](#) serta spesifikasi [jenis instans Amazon EC2 di Panduan Jenis Instans Amazon EC2](#).

Instans generasi saat ini

- Tujuan umum: M5 | M5a | M5ad | M5d | M5dn | M5n | M5zn | M6a | M6g | M6gd | M6i | M6iD | M6idn | M6in | M7a | M7g | M7gd | M7i | M7i-Flex | Mac1 | Mac2 | Mac2 | Mac2-m2 | Mac2-M2Pro | T2 | T3 | T3a | T4G
- Komputasi dioptimalkan: C5 | C5a | C5ad | C5d | C5n | C6a | C6g | C6gD | C6gN | C6i | C6iD | C6in | C7a | C7g | C7gd | C7gN | C7i | C7i-flex
- Memori dioptimalkan: R5 | R5a | R5ad | R5b | R5d | R5dn | R5n | R6a | R6g | R6gd | R6i | R6iDn | R6in | R6iD | R7a | R7g | R7gd | R7i | R7iZ | U-3tb1 | U-6tb1 | U-9TB1 TB1 | U-12TB1 | U-18TB1 | U-24tb1 | U7i-12TB | U7in-16TB | U7in-24TB | U7in-32tb | X1 | x2GD | x2idn | X2iEDN | X2IEZN | X1e | z1d
- Penyimpanan dioptimalkan: D2 | D3 | D3en | H1 | I3 | i3en | i4G | i4i | IM4GN | IS4gen
- Komputasi yang dipercepat: DL1 | DL2q | F1 | G4ad | G4dn | G5 | G5g | G6 | Gr6 | Inf1 | Inf2 | P2 | P3 | P3dn | P4d | P4de | P5 | Trn1 | Trn1n | VT1
- Komputasi kinerja tinggi: hPC6a | hPC6id | hPC7a | hPC7g

Instans generasi sebelumnya

- Tujuan umum: A1 | M1 | M2 | M3 | M4 | T1
- Komputasi dioptimalkan: C1 | C3 | C4
- Memori dioptimalkan: R3 | R4
- Penyimpanan dioptimalkan: I2
- Komputasi yang dipercepat: G3

Spesifikasi perangkat keras

Untuk spesifikasi tipe instans terperinci, lihat [Spesifikasi](#) di Panduan Jenis Instans Amazon EC2. Untuk informasi harga, lihat Harga Sesuai [Permintaan Amazon EC2](#).

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 [Ubah tipe instans](#).

Fitur prosesor Intel

Instans Amazon EC2 yang berjalan pada prosesor Intel dapat mencakup fitur-fitur berikut. Tidak semua fitur prosesor berikut didukung oleh semua tipe instans. Untuk informasi mendetail tentang fitur yang tersedia untuk setiap jenis instans, lihat Jenis [Instans Amazon EC2](#).

- Instruksi Baru Intel AES (AES-NI) — Kumpulan instruksi enkripsi Intel AES-NI melakukan peningkatan berdasarkan algoritma Advanced Encryption Standard (AES) asli untuk memberikan perlindungan data yang lebih cepat dan keamanan yang lebih kuat. Semua instans EC2 generasi saat ini mendukung fitur prosesor ini.
- Ekstensi Vektor Tingkat Lanjut Intel (Intel AVX, Intel AVX2, dan Intel AVX-512) — Intel AVX dan Intel AVX2 adalah 256-bit, dan Intel AVX-512 adalah ekstensi kumpulan instruksi 512-bit yang dirancang untuk aplikasi yang intensif Floating Point (FP). Instruksi Intel AVX meningkatkan performa untuk aplikasi, seperti pemrosesan gambar dan audio/video, simulasi ilmiah, analitik keuangan, serta 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 Gen ke-2 memperluas Intel AVX-512 dengan Instruksi Jaringan Neural Vektor (VNNI/INT8) baru yang secara signifikan meningkatkan performa inferensi deep learning dibandingkan prosesor Intel Xeon Scalable generasi sebelumnya (dengan FP32) untuk pengenalan/segmentasi gambar, deteksi objek, pengenalan ucapan, penerjemahan 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.

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 lebih lanjut, lihat [Memulai dengan Graviton](#).

AWS Trainium

Instans yang didukung oleh [AWS Trainium](#) dibuat khusus untuk pelatihan pembelajaran mendalam yang berkinerja tinggi dan hemat biaya. Anda dapat menggunakan contoh ini untuk melatih pemrosesan bahasa alami, visi komputer, dan model pemberi rekomendasi yang digunakan di serangkaian aplikasi yang luas, 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 Inferensia

Instans yang didukung oleh [AWS Inferentia](#) dirancang untuk mempercepat pembelajaran mesin. Mereka memberikan inferensi pembelajaran mesin berkinerja 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, layanan yang dikelola sepenuhnya yang merupakan cara termudah untuk memulai dengan model pembelajaran mesin. Untuk informasi selengkapnya, lihat [Memulai SageMaker](#) di Panduan SageMaker Pengembang Amazon.

- Luncurkan instans Inf1 atau Inf2 menggunakan AMI Deep Learning. Untuk informasi selengkapnya, lihat [AWS Inferentia dengan DLAMI](#) di Panduan Developer AWS Deep Learning AMI .
- 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 instans kontainer menggunakan instans Inf1 atau Inf2 dan AMI yang dioptimalkan Amazon ECS. Untuk informasi selengkapnya, lihat [AMI Amazon Linux 2 \(Inferentia\)](#) di Panduan Developer Amazon Elastic Container Service.
- Buat kluster Amazon EKS dengan simpul yang menjalankan instans Inf1. Untuk informasi selengkapnya, lihat [dukungan Inferentia](#) di Panduan Pengguna Amazon EKS.

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 Wilayah mendukung instance PV. Untuk informasi selengkapnya, lihat [Tipe virtualisasi AMI](#).

Untuk mendapatkan performa terbaik, kami menyarankan Anda menggunakan AMI HVM. Selain itu, AMI HVM diperlukan untuk memanfaatkan jaringan yang ditingkatkan. Virtualisasi HVM menggunakan teknologi hardware-assist 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.

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.

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 “Untuk beban kerja yang ada”](#).

Tugas

- [Untuk menemukan tipe instans menggunakan konsol](#)
- [Temukan jenis instance menggunakan AWS CLI](#)

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 tipe instans yang akan ditampilkan, seperti harga Linux Sesuai Permintaan, lalu pilih Konfirmasi. 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 melihat side-by-side perbandingan, pilih kotak centang untuk beberapa jenis 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Temukan jenis instance menggunakan AWS CLI

Anda dapat menggunakan AWS CLI perintah untuk Amazon EC2 untuk menemukan jenis instans yang memenuhi kebutuhan Anda.

Untuk menemukan jenis instance menggunakan AWS CLI

1. Jika Anda belum melakukannya, instal AWS CLI Untuk informasi lebih lanjut, lihat [Panduan AWS Command Line Interface Pengguna](#).
2. Gunakan perintah [description-instans-types](#) untuk memfilter tipe instans berdasarkan atribut instans. Misalnya, Anda dapat menggunakan perintah berikut untuk hanya menampilkan tipe instans generasi saat ini dengan memori 64 GiB (65.536 MiB).

```
aws ec2 describe-instance-types --filters "Name=current-generation,Values=true"
"Name=memory-info.size-in-mib,Values=65536" --query "InstanceTypes[*].
[InstanceType]" --output text | sort
```

3. Gunakan perintah [describe-instans-type-offerings](#) untuk memfilter tipe instans yang ditawarkan oleh lokasi (Wilayah atau Zone). Misalnya, Anda dapat menggunakan perintah berikut untuk menampilkan tipe instans yang ditawarkan di Zone 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
```

4. Setelah menemukan tipe instans yang memenuhi kebutuhan Anda, simpan daftarnya sehingga Anda dapat menggunakan tipe instans ini saat meluncurkan instans. Untuk informasi selengkapnya, lihat [Meluncurkan instans Anda](#) di Panduan Pengguna AWS Command Line Interface .

Mendapatkan rekomendasi untuk tipe instans

Alat berikut dapat membantu Anda memilih tipe instans optimal untuk beban kerja baru atau lama:

- **Beban kerja baru** — Pencari tipe instans EC2 mempertimbangkan kasus penggunaan Anda, jenis beban kerja, preferensi produsen CPU, dan bagaimana 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.
- **Beban kerja yang ada** — AWS Compute Optimizer menganalisis spesifikasi instans dan metrik pemanfaatan yang ada. Kemudian menggunakan data yang dikompilasi untuk merekomendasikan tipe instans Amazon EC2 mana yang dioptimalkan untuk biaya atau performa, atau keduanya, untuk beban kerja Anda yang ada.

Dapatkan rekomendasi tipe instans:

- [Dapatkan rekomendasi tipe instans untuk beban kerja baru](#)
- [Dapatkan rekomendasi tipe instans untuk beban kerja yang sudah ada](#)

Dapatkan rekomendasi tipe instans untuk beban kerja baru

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 dapat memakan waktu dan kompleks. Dengan menggunakan pencari tipe instans EC2, Anda dapat tetap up to date dengan jenis instans terbaru dan mencapai kinerja harga terbaik untuk beban kerja Anda.

Topik ini menguraikan cara mendapatkan saran dan panduan untuk jenis instans EC2 melalui 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 tipe instans untuk beban kerja yang sudah ada](#).

Gunakan pencari tipe instans EC2

Di konsol Amazon EC2, Anda bisa mendapatkan saran jenis instans dari pencari jenis 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 jenis instans EC2 di konsol Amazon EC2. Untuk melihat animasi langkah-langkahnya, lihat [Lihat animasi: Dapatkan saran tipe instans menggunakan pencari tipe instans EC2](#).

Untuk mendapatkan saran tipe 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 Instance, lalu pilih tombol Instance type finder.
2. Di layar Dapatkan saran tentang pemilihan jenis instance, lakukan hal berikut:
 - a. Tentukan persyaratan jenis instans Anda dengan memilih opsi untuk jenis Beban Kerja, Kasus penggunaan, Prioritas, 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 Tambah, 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 untuk keluarga misalnya yang cocok 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 Actions, Launch instance atau Actions, Create launch template.

Atau, jika Anda memulai proses di wizard instance peluncuran atau halaman template peluncuran, dan Anda lebih suka kembali ke alur asli Anda, catat jenis instance yang ingin Anda gunakan. Kemudian, di wizard instance peluncuran atau template peluncuran, untuk jenis Instance, pilih jenis instance, dan selesaikan prosedur untuk meluncurkan instance atau membuat template peluncuran.

Lihat animasi: Dapatkan saran tipe instans menggunakan pencari tipe instans EC2

The screenshot displays the AWS Management Console interface for EC2 resources in the US East (N. Virginia) Region. The left sidebar shows navigation options like EC2 Dashboard, EC2 Global View, Events, and various instance-related categories. The main content area is divided into several sections:

- Resources:** A table showing the current usage of various EC2 resources:

Resource Type	Count
Instances (running)	2
Dedicated Hosts	0
Instances	2
Load balancers	0
Security groups	12
Volumes	2
Auto Scaling Groups	0
Elastic IPs	0
Key pairs	0
Placement groups	0
Snapshots	3
- Launch instance:** A section with a "Launch Instance" button and a "Migrate a server" link. A note states: "Note: Your instances will launch in the US East (N. Virginia) Region".
- Service health:** Shows the "AWS Health Dashboard" for the region "US East (N. Virginia)" with a status of "This service is operating normally."
- Account attributes:** Displays the "Default VPC" (vpc-92304aeb) and various settings like "Data protection and security", "Zones", "EC2 Serial Console", "Default credit specification", and "EC2 console preferences".
- Explore AWS:** Promotes performance improvements, such as "Get Up to 40% Better Price Performance" for T4g instances and "Enable Best Price-Performance with AWS Graviton2".

Dapatkan rekomendasi tipe instans untuk beban kerja yang sudah ada

AWS Compute Optimizer memberikan rekomendasi instans Amazon EC2 untuk membantu Anda meningkatkan kinerja, menghemat uang, atau keduanya. Anda dapat menggunakan rekomendasi ini untuk memutuskan apakah akan beralih ke tipe instans yang 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.

Topik ini menguraikan cara melihat rekomendasi melalui konsol Amazon EC2. Untuk informasi selengkapnya, lihat [Panduan Pengguna.AWS Compute Optimizer](#)

Note

Untuk mendapatkan rekomendasi dari Compute Optimizer, Anda harus terlebih dahulu memilih Compute Optimizer. Untuk informasi selengkapnya, lihat [Memulai AWS Compute Optimizer](#) dalam Panduan Pengguna AWS Compute Optimizer .

Jika Anda mencari rekomendasi tipe instans untuk beban kerja baru, gunakan pemilih tipe instans Amazon Q EC2. Untuk informasi selengkapnya, lihat [Dapatkan rekomendasi tipe instans untuk beban kerja baru](#).

Daftar Isi

- [Batasan](#)
- [Temuan](#)
- [Melihat rekomendasi](#)
- [Pertimbangan untuk mengevaluasi rekomendasi](#)
- [Sumber daya tambahan](#)

Batasan

Compute Optimizer saat ini menghasilkan rekomendasi untuk tipe instans C, D, H, I, M, R, T, X, dan z. Tipe instans lain tidak dianggap oleh Compute Optimizer. Jika Anda menggunakan tipe instans lain, tipe instans tersebut tidak akan terdaftar di tampilan rekomendasi Compute Optimizer. Untuk informasi selengkapnya tentang tipe instans yang didukung dan yang tidak didukung, lihat [Persyaratan instans Amazon EC2](#) dalam Panduan Pengguna AWS Compute Optimizer .

Temuan

Compute Optimizer mengklasifikasikan temuannya untuk instans EC2 sebagai berikut:

- Penyediaan tidak mencukupi - Instans EC2 dianggap tidak mencukupi ketika setidaknya satu spesifikasi instans Anda, seperti CPU, memori, atau jaringan, tidak memenuhi persyaratan performa beban kerja Anda. Instans EC2 yang tidak mencukupi dapat menyebabkan performa aplikasi yang buruk.

- Disediakan secara berlebihan – Sebuah instans EC2 dianggap disediakan secara berlebihan ketika setidaknya satu spesifikasi dari instans Anda, seperti CPU, memori, atau jaringan, dapat diturunkan ukurannya tetapi masih memenuhi persyaratan performa beban kerja Anda, dan ketika tidak ada spesifikasi yang kurang disediakan. Instans EC2 yang disediakan secara berlebihan 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. Untuk informasi selengkapnya, lihat [Batasan](#) di bagian sebelumnya.

Melihat rekomendasi

Setelah memilih Compute Optimizer, Anda dapat melihat temuan yang dihasilkan oleh Compute Optimizer untuk instans EC2 Anda di konsol 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 instans EC2 melalui konsol EC2

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans, lalu pilih ID instans .
3. Di halaman ringkasan instans, di banner AWS Compute Optimizer di sekitar bagian bawah halaman, pilih Lihat detail.

Instans terbuka di Compute Optimizer, yang labeli sebagai instans Saat Ini. Tersedia hingga tiga rekomendasi tipe instans yang berbeda, berlabel Opsi 1, Opsi 2, dan Opsi 3. Bagian bawah jendela menunjukkan data CloudWatch metrik terbaru untuk contoh saat ini: pemanfaatan CPU, pemanfaatan Memori, Jaringan masuk, dan Jaringan keluar.

4. (Opsional) Di konsol Compute Optimizer, pilih settings



) untuk mengubah kolom yang terlihat dalam tabel, atau untuk melihat informasi harga publik untuk opsi pembelian yang berbeda untuk jenis instans saat ini dan yang direkomendasikan.

 Note

Jika Anda telah membeli Instans Terpesan, Instans Sesuai Permintaan Anda mungkin ditagih sebagai Instans Terpesan. Sebelum Anda mengubah tipe instans saat ini, pertama-tama evaluasi dampaknya terhadap penggunaan dan cakupan Instans Terpesan.

Tentukan apakah Anda ingin menggunakan salah satu rekomendasi. Tentukan apakah akan mengoptimalkan peningkatan performa, pengurangan biaya, atau kombinasi keduanya. Untuk informasi selengkapnya, lihat [Melihat Rekomendasi Sumber Daya](#) dalam Panduan Pengguna AWS Compute Optimizer .

Untuk melihat rekomendasi semua instans EC2 di seluruh Wilayah melalui konsol Compute Optimizer

1. Buka konsol Compute Optimizer di <https://console.aws.amazon.com/compute-optimizer/>.
2. Pilih Lihat rekomendasi untuk semua instans EC2.
3. Anda dapat melakukan tindakan berikut di halaman rekomendasi:
 - a. Untuk memfilter rekomendasi ke satu atau beberapa AWS Wilayah, masukkan nama Wilayah di kotak teks Filter menurut satu atau beberapa Wilayah, atau pilih satu atau beberapa Wilayah dalam daftar drop-down yang muncul.
 - b. Untuk melihat rekomendasi sumber daya di akun lain, pilih Akun, lalu pilih ID akun yang berbeda.

Opsi ini tersedia hanya jika Anda masuk ke akun manajemen organisasi, dan Anda memilih di semua akun anggota dalam organisasi.

- c. Untuk menghapus filter yang dipilih, pilih Hapus filter.
- d. Untuk mengubah opsi pembelian yang ditampilkan untuk jenis instans saat ini dan yang direkomendasikan, pilih pengaturan



),
lalu pilih Instans Sesuai Permintaan, Instans Cadangan, standar 1 tahun tanpa dimuka, atau Instans Cadangan, standar 3 tahun tanpa dimuka.

- e. Untuk melihat detail, seperti rekomendasi tambahan dan perbandingan metrik pemanfaatan, pilih temuan (Kurang mencukupi, Berlebihan, atau Dioptimalkan) yang tercantum di samping instans yang diinginkan. Untuk informasi selengkapnya, lihat [Melihat Detail Sumber Daya](#) dalam Panduan Pengguna AWS Compute Optimizer .

Pertimbangan untuk mengevaluasi rekomendasi

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 utama dan definisi untuk instans performa yang dapat melonjak](#).
- Jika Anda telah membeli Instans Terpesan, Instans Sesuai Permintaan Anda mungkin ditagih sebagai Instans Terpesan. 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 keluarkan untuk memvalidasi apakah tipe 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.

Ada pertimbangan lain saat mengubah ukuran instans EC2. Untuk informasi selengkapnya, lihat [Ubah tipe instans](#).

Sumber daya tambahan

Untuk informasi selengkapnya:

- [Jenis Instans Amazon EC2](#)
- [AWS Compute Optimizer Panduan Pengguna](#)

Ubah tipe instans

Saat kebutuhan Anda berubah, Anda mungkin menemukan bahwa instans Anda digunakan secara berlebihan (tipe instans terlalu kecil) atau kurang termanfaatkan (tipe instans terlalu besar). 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 tipe instans untuk beban kerja yang sudah ada](#).

Saat mengubah tipe instans, Anda akan mulai membayar tarif tipe instans yang baru. Untuk tarif sesuai permintaan semua tipe instans, lihat: [Harga Sesuai Permintaan Amazon EC2](#).

Untuk menambahkan penyimpanan tambahan ke instans Anda tanpa mengubah jenis instans, tambahkan volume EBS ke instance. Untuk informasi selengkapnya, lihat [Melampirkan 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 jenis instans dari instans yang didukung EBS
EBS	Tidak kompatibel	Ubah tipe instans dengan meluncurkan instans baru
Penyimpanan instans	Tidak berlaku	Ubah tipe instans dari instans yang didukung penyimpanan instans

Pertimbangan untuk tipe instans yang kompatibel

Pertimbangkan hal berikut saat mengubah tipe instans dari instans yang ada:

- Anda harus menghentikan instans yang didukung Amazon EBS sebelum dapat mengubah tipe 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 [Hentikan dan mulai instans Amazon EC2](#).
- Saat Anda berhenti dan memulai sebuah instans, kami memindahkan instans tersebut ke perangkat keras baru. Jika instans Anda memiliki alamat IPv4 publik, kami merilis alamat dan memberi instans Anda alamat IPv4 yang baru. Jika Anda memerlukan alamat IPv4 publik yang tidak berubah, gunakan [alamat IP Elastis](#).
- Anda tidak dapat mengubah tipe instans dari [Instans Spot](#).
- [Instans Windows] Kami menyarankan Anda memperbarui paket driver AWS PV sebelum mengubah jenis instans. Untuk informasi selengkapnya, lihat [the section called “Mutakhirkan driver PV”](#).
- Jika instans Anda berada dalam grup Auto Scaling, layanan Amazon EC2 Auto Scaling menandai instans yang dihentikan sebagai tidak sehat, dan dapat mengakhiri instans tersebut serta meluncurkan instans 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 [Batasan volume instans](#).

Ubah jenis instans dari instans yang didukung EBS

Gunakan petunjuk berikut untuk mengubah tipe instans dari instans yang didukung EBS jika tipe instans yang Anda butuhkan kompatibel dengan konfigurasi instans saat ini.

Untuk mengubah tipe instans dari instans yang didukung Amazon EBS

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 instance Windows Anda untuk menggunakan [pengalamatan IP statis](#) dan Anda mengubah dari jenis instans yang tidak mendukung jaringan yang ditingkatkan ke jenis instans yang mendukung jaringan yang ditingkatkan, Anda mungkin mendapatkan peringatan tentang potensi konflik alamat IP saat Anda mengkonfigurasi ulang pengalamatan IP statis. Untuk mencegahnya, aktifkan DHCP pada antarmuka jaringan untuk instans Anda sebelum Anda mengubah tipe instans. Dari instans Anda, buka Pusat Jaringan dan Berbagi, buka Properti Protokol Internet Versi 4 (TCP/IPv4) untuk antarmuka jaringan, dan pilih Dapatkan alamat IP secara otomatis. Ubah tipe instans dan konfigurasi kembali pengalamatan IP statis pada antarmuka jaringan.
3. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
4. Di panel navigasi, pilih Contoh.
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: [Ubah tipe instans dengan meluncurkan instans baru](#).

- b. (Opsional) Jika tipe instans yang Anda pilih mendukung pengoptimalan EBS, pilih EBS dioptimalkan untuk mengaktifkan pengoptimalan EBS, atau batalkan pilihan EBS dioptimalkan untuk menonaktifkan pengoptimalan EBS. Jika tipe instans yang Anda pilih adalah EBS – dioptimalkan secara default, EBS-dioptimalkan dipilih dan Anda tidak dapat membatalkan pilihannya.
 - c. Pilih Terapkan 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 instance 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 [Mengkonfigurasi tugas inisialisasi](#) (EC2Launch) atau [Ubah pengaturan](#) (EC2Launch v2).

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

Ubah tipe instans dengan meluncurkan instans baru

Jika konfigurasi instans yang didukung EBS Anda tidak kompatibel dengan tipe instans baru yang diinginkan, Anda tidak dapat mengubah tipe instans asli. Sebagai gantinya, Anda harus meluncurkan instans baru dengan konfigurasi yang kompatibel dengan tipe instans baru yang Anda inginkan, dan kemudian memigrasikan aplikasi Anda ke instans baru. Misalnya, jika Anda meluncurkan instans

asli dari AMI PV, tetapi ingin mengubah ke jenis instans generasi saat ini yang memerlukan AMI HVM, Anda harus meluncurkan instance baru dari AMI HVM. Untuk informasi tentang bagaimana kompatibilitas ditentukan, lihat [Kompatibilitas untuk mengubah tipe instans](#).

Untuk memigrasikan aplikasi Anda ke instans baru, lakukan hal berikut:

- Cadangkan data pada instans asli Anda.
- Luncurkan instans baru dengan konfigurasi yang kompatibel dengan tipe instans baru yang Anda inginkan, dan lampirkan volume EBS apa pun yang dilampirkan ke instans asli Anda.
- Instal aplikasi Anda dan perangkat lunak apa pun pada instans baru.
- Pulihkan data apa pun.
- Jika instans asli Anda memiliki alamat IP Elastis, dan Anda ingin memastikan bahwa pengguna Anda dapat melanjutkan tanpa gangguan untuk menggunakan aplikasi pada instans baru Anda, Anda harus mengaitkan alamat IP Elastis dengan instans baru Anda. Untuk informasi selengkapnya, lihat [Alamat IP Elastis](#).

Untuk mengubah tipe instans untuk konfigurasi instans baru

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Cadangkan data yang perlu Anda simpan, sebagai berikut:
 - Untuk data dalam volume penyimpanan instans Anda, cadangkan data ke penyimpanan persisten.
 - Untuk data pada volume EBS Anda, buat snapshot volume atau lepaskan volume dari instance sehingga Anda dapat melampirkannya ke instance baru nanti.
3. Di panel navigasi, pilih Instans.
4. Pilih Luncurkan Instans. Saat Anda mengonfigurasi instans, lakukan hal berikut:
 - a. Pilih AMI yang mendukung tipe instans yang Anda inginkan. Perhatikan bahwa tipe 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 menggunakan alamat IP Elastis, pilih VPC tempat instans asli saat ini berjalan.
 - d. Jika Anda ingin mengizinkan lalu lintas yang sama untuk mencapai instans baru, pilih grup keamanan yang terkait dengan instans asli.

- e. Saat Anda selesai mengonfigurasi instans baru, selesaikan langkah-langkah untuk memilih pasangan kunci dan meluncurkan instans Anda. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`.
5. Jika diperlukan, lampirkan volume EBS baru berdasarkan snapshot yang Anda buat, atau volume EBS yang Anda lepaskan dari instans asli, ke instans 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 Anda menggunakan alamat IP Elastis, tetapkan ke instans 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 Tindakan, Kaitkan alamat IP Elastis.
 - d. Untuk tipe Resource, pilih instans.
 - e. Untuk Instans, pilih instans baru yang akan dikaitkan dengan alamat IP Elastis.
 - 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.

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.

[Instance Linux] Anda dapat menggunakan `AWSSupport-MigrateXenToNitroLinux` runbook untuk memigrasikan instance Linux yang kompatibel dari jenis instance Xen ke jenis instans Nitro. Untuk informasi selengkapnya, lihat [AWSSupport-MigrateXenToNitroLinux runbook](#) di referensi buku runbook Otomatisasi AWS Systems Manager .

[Instans Windows] Untuk panduan tambahan tentang memigrasi instance Windows yang kompatibel dari tipe instans Xen ke tipe instans Nitro, lihat [Memigrasi](#) ke jenis instans generasi terbaru.

Kompatibilitas ditentukan dengan cara 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 AMI](#). 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)

Kartu jaringan

Beberapa tipe instans mendukung banyak [kartu jaringan](#). Anda harus memilih tipe instans yang mendukung jumlah kartu jaringan yang sama dengan tipe instans saat ini.

Jaringan yang ditingkatkan

Tipe instans yang mendukung [jaringan yang ditingkatkan](#) memerlukan instalasi driver yang diperlukan. Misalnya, [instans yang dibangun di Sistem AWS Nitro](#) memerlukan AMI yang didukung EBS dengan driver Elastic Network Adapter (ENA) yang 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 [instans yang dibangun di](#) Sistem Nitro. AWS Jika Anda mengubah dari tipe instans yang tidak mendukung NVMe ke tipe instans yang mendukung NVMe, Anda harus menginstal driver NVMe pada instans Anda. Selain itu, nama perangkat untuk perangkat yang Anda tentukan dalam pemetaan perangkat blok diganti namanya menggunakan nama perangkat NVMe (`/dev/nvme[0-26]n1`

[Instance Linux] Oleh karena itu, untuk me-mount sistem file saat boot menggunakan `/etc/fstab`, Anda harus menggunakan UUID/label alih-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 [Batasan volume instans](#).

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.

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 kluster

Jika instans Anda berada dalam [grup penempatan kluster](#) dan, setelah mengubah tipe instans, instans gagal dimulai, coba yang berikut ini:

1. Hentikan semua instans dalam grup penempatan kluster.
2. Mengubah tipe instans yang terpengaruh.
3. Mulai semua instans dalam grup penempatan kluster.

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 instans berhenti dan dimulai, kami menyarankan Anda menggunakan alamat IP Elastis, tanpa biaya tambahan asalkan instans Anda berjalan. Untuk informasi selengkapnya, lihat [Alamat IP elastis](#).

Ubah tipe instans dari instans yang didukung penyimpanan instans

Instans yang didukung oleh penyimpanan instans adalah instans yang memiliki volume root penyimpanan instans. Anda tidak dapat mengubah tipe instans yang memiliki volume root penyimpanan instans. Sebagai gantinya, Anda harus membuat AMI dari instans Anda, meluncurkan instans baru dari AMI ini dan memilih tipe instans yang Anda inginkan, lalu memigrasikan aplikasi Anda ke instans baru. Perhatikan bahwa tipe instans yang Anda inginkan harus kompatibel dengan AMI yang Anda buat. Untuk informasi tentang bagaimana kompatibilitas ditentukan, lihat [Kompatibilitas untuk mengubah tipe instans](#).

Gambaran umum proses

- Cadangkan data pada instans asli Anda.
- Buat AMI dari instans asli Anda.
- Luncurkan instans baru dari AMI ini dan pilih tipe instans yang Anda inginkan.
- Instal aplikasi Anda pada instans baru.
- Jika instans asli Anda memiliki alamat IP Elastis, dan Anda ingin memastikan bahwa pengguna Anda dapat melanjutkan tanpa gangguan untuk menggunakan aplikasi pada instans baru Anda, Anda harus mengaitkan alamat IP Elastis dengan instans baru Anda. Untuk informasi selengkapnya, lihat [Alamat IP Elastis](#).

Untuk mengubah tipe instans dari instans yang didukung penyimpanan instans

1. Cadangkan data yang perlu Anda simpan, sebagai berikut:
 - Untuk data dalam volume penyimpanan instans Anda, cadangkan data ke penyimpanan persisten.
 - Untuk data pada volume EBS Anda, buat snapshot volume atau lepaskan volume dari instance sehingga Anda dapat melampirkannya ke instance baru nanti.
2. Buat AMI dari instans Anda dengan memenuhi prasyarat dan mengikuti prosedur di [Buat AMI Linux yang didukung penyimpanan instans](#). Saat Anda selesai membuat AMI dari instans Anda, kembali ke prosedur ini.

3. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
4. Di panel navigasi, pilih AMI. Dari daftar filter, pilih Milik saya, lalu pilih citra yang Anda buat di Langkah 2. Perhatikan bahwa Nama AMI adalah nama yang Anda tentukan saat Anda mendaftarkan gambar dan Sumber adalah bucket Amazon S3 Anda.

 Note

Jika Anda tidak melihat AMI yang Anda buat di Langkah 2, pastikan Anda telah memilih Wilayah tempat Anda membuat AMI.

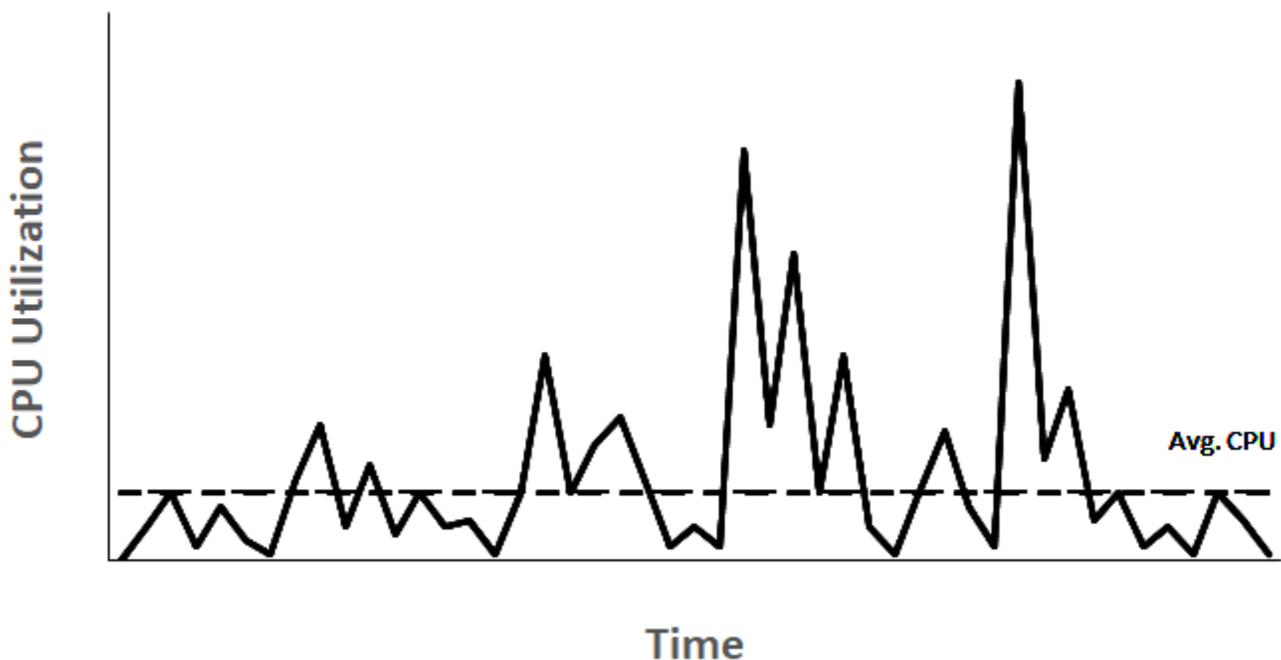
5. Dengan AMI yang dipilih, pilih Luncurkan instans dari gambar. Saat Anda mengonfigurasi instans, lakukan hal berikut:
 - a. Pilih tipe instans baru yang Anda inginkan. Jika tipe instans yang Anda inginkan tidak tersedia, maka instans itu tidak kompatibel dengan konfigurasi AMI yang Anda buat. Untuk informasi selengkapnya, lihat [Kompatibilitas untuk mengubah tipe instans](#).
 - b. Jika Anda menggunakan alamat IP Elastis, pilih VPC tempat instans asli saat ini berjalan.
 - c. Jika Anda ingin mengizinkan lalu lintas yang sama untuk mencapai instans baru, pilih grup keamanan yang terkait dengan instans asli.
 - d. Saat Anda selesai mengonfigurasi instans baru, selesaikan langkah-langkah untuk memilih pasangan kunci dan meluncurkan instans Anda. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`.
6. Jika diperlukan, lampirkan volume EBS baru berdasarkan snapshot yang Anda buat, atau volume EBS yang Anda lepaskan dari instans asli, ke instans baru.
7. Instal aplikasi Anda dan perangkat lunak yang diperlukan pada instans baru.
8. Jika Anda menggunakan alamat IP Elastis, tetapkan ke instans 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 Tindakan, Kaitkan alamat IP Elastis.
 - d. Untuk tipe Resource, pilih instans.
 - e. Untuk Instans, pilih instans baru yang akan dikaitkan dengan alamat IP Elastis.
 - 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.

Instans 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 Cloud saat AWS ini.

Many common workloads look like this



Beban kerja pemanfaatan low-to-moderate CPU ini menyebabkan pemborosan siklus CPU dan, akibatnya, Anda membayar lebih dari yang Anda gunakan. 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 komputasi, memori, dan sumber daya jaringan, dan memberi Anda cara yang paling hemat biaya untuk menjalankan spektrum luas aplikasi tujuan umum yang memiliki penggunaan CPU. low-to-moderate 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

Tipe instans yang dapat melonjak EC2 terdiri dari tipe instans T4g, T3a dan T3, serta 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 instans T4G didukung oleh prosesor [AWS Graviton2](#) berbasis ARM dengan dukungan ekosistem yang luas dari vendor sistem operasi, vendor perangkat lunak independen, dan layanan dan aplikasi populer. AWS

Tabel berikut merangkum perbedaan utama antara tipe-tipe instans yang dapat melonjak.

Tipe	Deskripsi	Keluarga prosesor
Generasi terbaru		
T4g	Tipe instans EC2 berbiaya terendah dengan harga/performa hingga 40% lebih tinggi dan biaya 20% lebih rendah vs. T3	AWS Prosesor Graviton2 dengan inti Arm Neoverse N1

Tipe	Deskripsi	Keluarga prosesor
T3a	Instans berbasis x86 paling murah dengan biaya 10% lebih rendah vs. instans T3	Prosesor EPYC generasi ke-1 dari AMD
T3	Harga/performa puncak terbaik untuk beban kerja x86 dengan harga/performa hingga 30% lebih rendah dibandingkan 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 akun Anda kurang dari 12 bulan, Anda dapat menggunakan instans `t2.micro` secara gratis (atau instans `t3.micro` di Wilayah tempat `t2.micro` tidak tersedia) dalam batas penggunaan tertentu. Untuk informasi selengkapnya, lihat [AWS Tingkat Gratis](#).

Opsi pembelian yang didukung untuk instans T

- Instans Sesuai Permintaan
- Instans Terpesan
- Instans Khusus (khusus T3)
- Host Khusus (khusus T3, hanya dalam mode standard)
- Instans Spot

Untuk informasi selengkapnya, lihat [Opsi pembelian instans](#).

Daftar Isi

- [Praktik terbaik](#)
- [Konsep utama dan definisi untuk instans performa yang dapat melonjak](#)
- [Mode tidak terbatas untuk instans performa yang dapat melonjak](#)
- [Mode standar untuk instans performa yang dapat melonjak](#)
- [Bekerja dengan instans performa yang dapat melonjak](#)
- [Pantau kredit CPU Anda untuk instans performa yang dapat melonjak](#)

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 utama dan definisi untuk instans performa yang dapat melonjak

Tipe instans Amazon EC2 tradisional menyediakan sumber daya CPU tetap, sementara instans performa yang dapat melonjak menyediakan tingkat pemanfaatan CPU dasar dengan kemampuan untuk melonjatkan 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 acuan, instans akan dikenai biaya untuk penggunaan tambahan dengan [tarif tambahan flat](#) per jam vCPU.

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 sedang dimanfaatkan pada instans. 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 AWS Management Console atau AWS CLI, lihat [Mendapatkan statistik untuk instans tertentu](#).

Kredit CPU

Satu unit vCPU-waktu.

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 acuan

Pemanfaatan acuan adalah tingkat di mana CPU dapat digunakan dengan saldo kredit bersih sebesar 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 % = (jumlah kredit yang didapatkan/jumlah 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:

$2 \text{ vCPU} * 5\% \text{ pemanfaatan dasar} * 60 \text{ menit} = 6 \text{ kredit per jam}$

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) = $6 - 2 \text{ vCPU} * 2\% \text{ pemanfaatan CPU} * 60 \text{ menit} = 6 - 2,4 = 3,6 \text{ kredit akumulasi per jam}$

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 = $24 * 6 = 144 \text{ kredit}$

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 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. Jika instans berjalan pada pemanfaatan CPU yang lebih tinggi untuk waktu yang lama, instans dapat melakukannya dengan [tarif tambahan tetap](#) per jam vCPU.

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 aktual, Kredit yang digunakan, Surplus kredit (hanya pada mode Tidak Terbatas)	Standar, Tidak Terbatas (default)	7 hari (kredit bertahan selama 7 hari setelah instans berhenti)

Tipe	Tipe kredit CPU yang didukung	Mode konfigurasi kredit	Masa pakai kredit CPU akumulasi antara instans mulai dan berhenti
T3a	Kredit yang diperoleh , Kredit aktual, 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 aktual, 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%**
t2.large	36	864	2	30%**
t2.xlarge	54	1296	4	22,5% **
t2.2xlarge	81.6	1958.4	8	17%**

Jenis instans	Kredit CPU didapatkan per jam	Kredit maksimum yang dihasilkan yang dapat dikumpulkan*	vCPU***	Pemanfaatan acuan per vCPU
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%**
t3a.2xlarge	192	4608	8	40%**
T4g				
t4g.nano	6	144	2	5%**

Jenis instans	Kredit CPU didapatkan per jam	Kredit maksimum yang dihasilkan yang dapat dikumpulkan*	vCPU***	Pemanfaatan acuan per vCPU
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%**

* Jumlah kredit yang dapat diperoleh setara dengan jumlah kredit yang bisa didapatkan dalam periode 24 jam.

** Pemanfaatan acuan persentase dalam tabel adalah per vCPU. Dalam CloudWatch, pemanfaatan CPU ditampilkan per vCPU. Misalnya, pemanfaatan CPU untuk t3.large instance yang beroperasi pada tingkat dasar ditampilkan sebagai 30% dalam CloudWatch metrik CPU. Untuk informasi tentang cara menghitung pemanfaatan dasar, lihat [Pemanfaatan acuan](#).

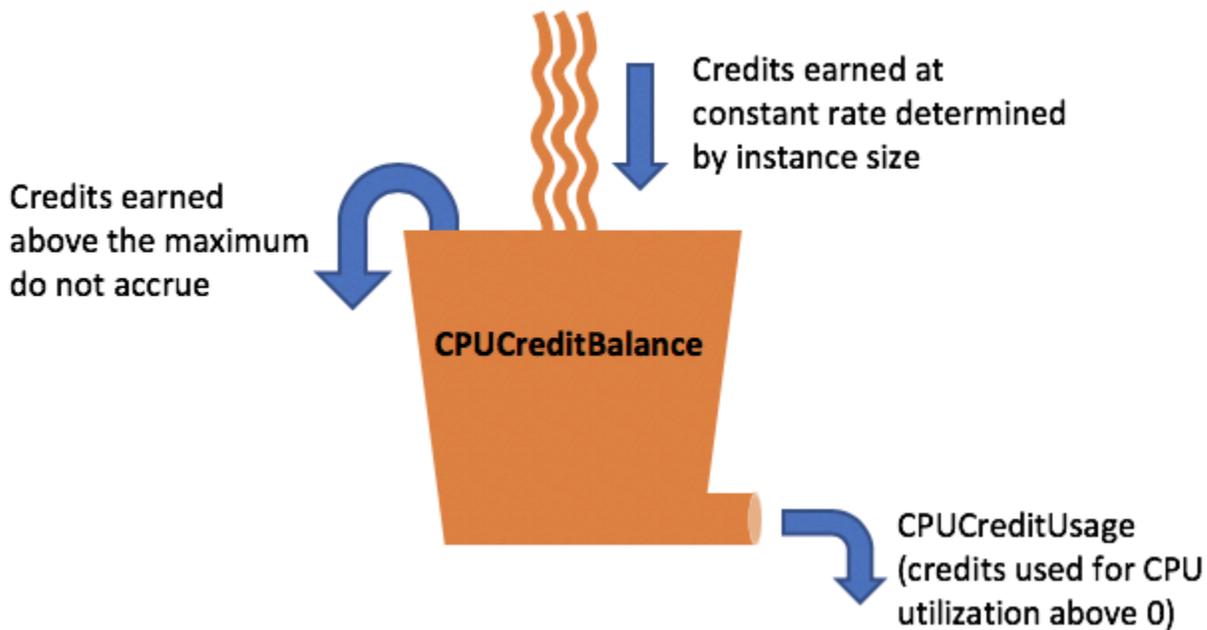
*** Tiap vCPU adalah thread dari inti Intel Xeon atau core AMD EPYC, kecuali untuk instans T2 dan T4g.

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 instans berhenti dan kredit hilang setelahnya. Jika Anda memulai instans dalam tujuh hari, tidak ada kredit yang hilang.

Untuk informasi selengkapnya, lihat `CPUCreditBalance` di [tabel CloudWatch metrik](#).

Pemanfaatan acuan

Pemanfaatan acuan adalah tingkat di mana CPU dapat digunakan dengan saldo kredit bersih sebesar 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:

$$\frac{(\text{number of credits earned}/\text{number of vCPUs})/60 \text{ minutes}}{1} = \% \text{ baseline utilization}$$

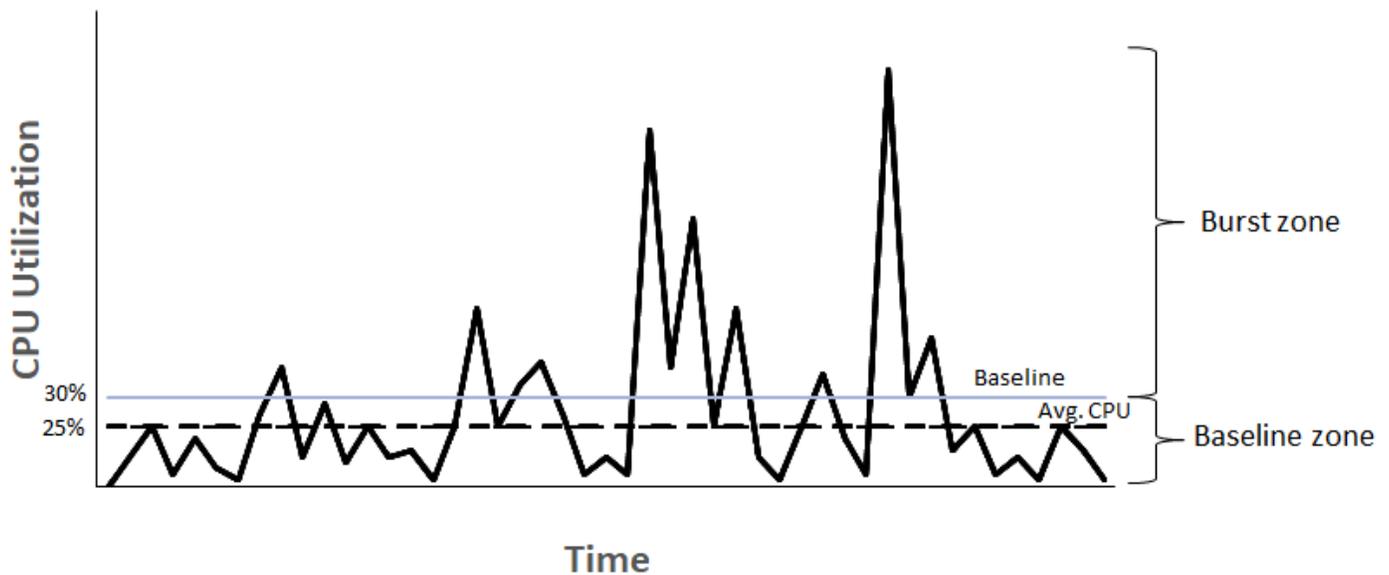
Misalnya, sebuah instans, `t3.nano` dengan 2 vCPU, mendapatkan 6 kredit per jam, menghasilkan pemanfaatan dasar 5%, yang dihitung sebagai berikut:

$$(6 \text{ credits earned}/2 \text{ vCPUs})/60 \text{ minutes} = 5\% \text{ baseline utilization}$$

Sebuah `t3.large` contoh, dengan 2 vCPU, menghasilkan 36 kredit per jam, menghasilkan pemanfaatan dasar 30% (). $(36/2)/60$

Grafik berikut memberikan contoh pemanfaatan CPU rata-rata di bawah baseline. `t3.large`

Example of t3.large



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 instans berjalan pada pemakaian CPU yang lebih tinggi untuk waktu yang lama, instans dapat melakukannya dengan tarif tambahan tetap per jam vCPU. Untuk informasi tentang harga, lihat [Harga Amazon EC2](#) dan [Harga Mode Tak Terbatas T2/T3/T4](#).

Jika Anda menggunakan instans `t2.micro` atau `t3.micro` berdasarkan penawaran [AWS Tingkat Gratis](#) dan menggunakannya dalam mode `unlimited`, maka biaya mungkin berlaku jika rata-rata pemanfaatan Anda selama periode 24 jam yang berputar melebihi [pemanfaatan dasar](#) instans.

[Instans T4G, T3a dan T3 diluncurkan sebagai unlimited default \(kecuali jika Anda mengubah default\)](#). 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 lebih lanjut, lihat [Kredit surplus dapat dikenakan biaya](#) dan [Instance performa yang dapat melonjak](#)

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 tidak terbatas](#)
 - [Cara kerja instans performa yang dapat melonjak Tidak Terbatas](#)
 - [Kapan menggunakan mode tak terbatas versus CPU tetap](#)
 - [Kredit surplus dapat dikenakan biaya](#)
 - [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 tidak terbatas](#)
 - [Contoh 1: Menjelaskan penggunaan kredit dengan T3 Tidak Terbatas](#)
 - [Contoh 2: Menjelaskan penggunaan kredit dengan T2 Tidak Terbatas](#)

Konsep mode tidak terbatas

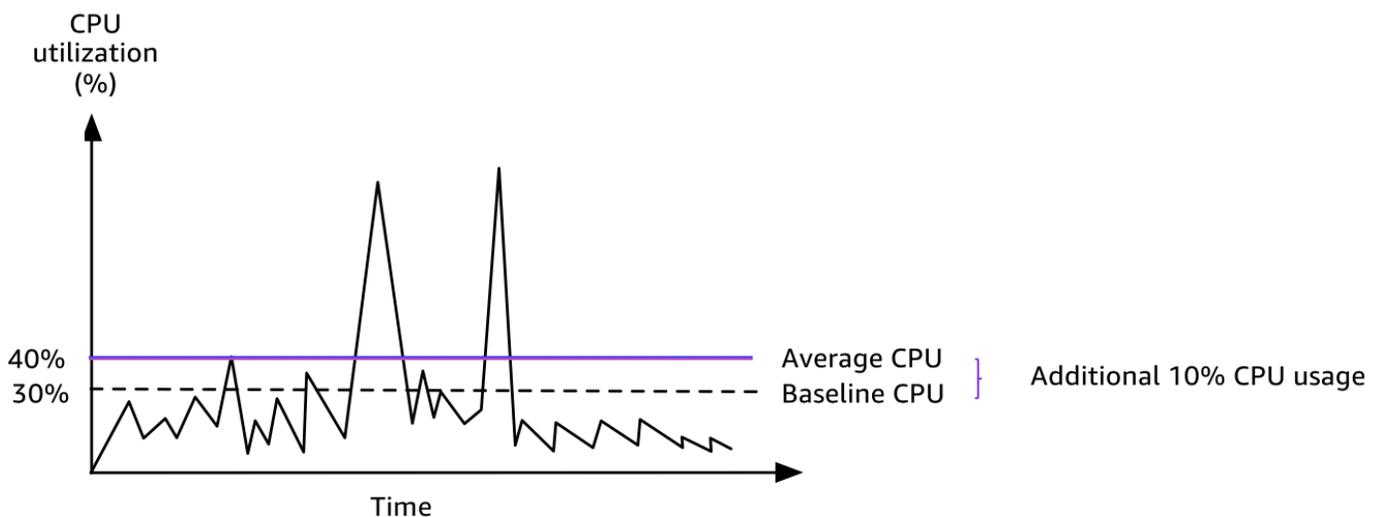
Mode `unlimited` adalah opsi konfigurasi kredit untuk instans performa yang dapat melonjak. Mode ini dapat diaktifkan atau dinonaktifkan kapan saja untuk instans yang berjalan atau dihentikan. Anda dapat [menetapkan `unlimited` sebagai opsi kredit default](#) di tingkat akun per AWS Wilayah, per keluarga instans performa burstable, sehingga semua instance performa 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 acuan, instans akan dikenai biaya untuk penggunaan tambahan dengan [tarif tambahan flat](#) per jam vCPU.

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 instans berjalan dengan rata-rata 40% pemanfaatan CPU selama 24 jam, seperti yang ditunjukkan dalam grafik, maka instans tersebut akan ditagih untuk 10% penggunaan CPU tambahan dengan [tarif tambahan tetap](#) per jam vCPU.



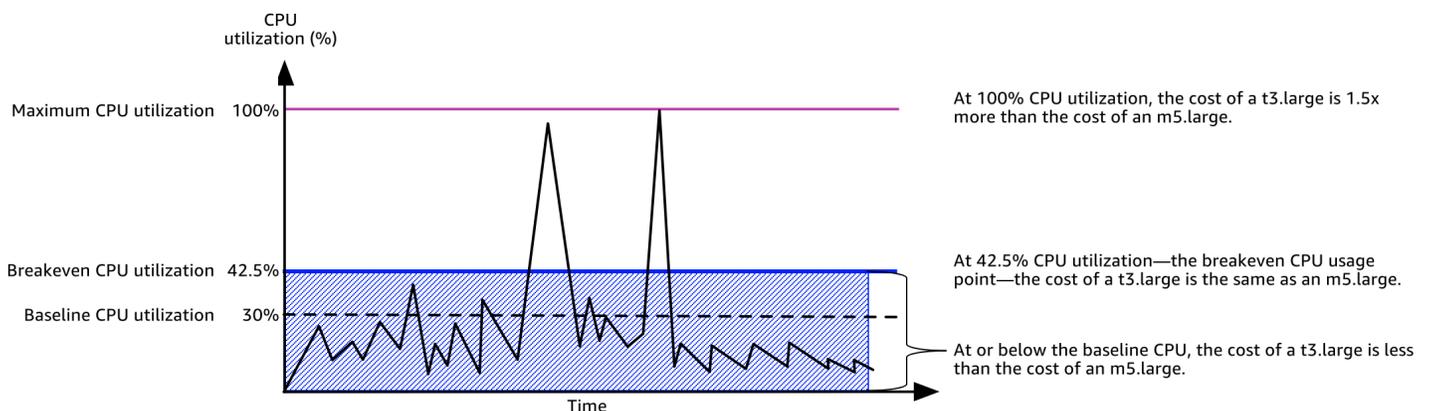
Untuk informasi selengkapnya tentang pemanfaatan acuan per vCPU untuk setiap tipe instans dan jumlah kredit yang didapatkan oleh setiap tipe instans, lihat [tabel kredit](#).

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 T3 per vCPU (%)	Biaya per vCPU untuk kredit surplus	Biaya per menit vCPU	Menit lonjakan tambahan per vCPU	% CPU tersedia	% CPU tambahan
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	\$0,000833	15	12,5%	42,5%

* 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 flat](#) per jam vCPU yang dikenakan pada instans jika melonjak ke 100% CPU setelah kredit yang diperolehnya habis.
- Kolom H menunjukkan [tarif tambahan flat](#) per jam vCPU-menit yang dikenakan pada instans jika melonjak ke 100% CPU setelah kredit yang diperolehnya 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
<code>t3.large</code>	42,5%
<code>t3.xlarge</code>	52,5%
<code>t3.2xlarge</code>	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 dibayarkan akan dikenakan tarif tambahan tetap per vCPU-jam. Untuk informasi tentang tarif, lihat [Harga Mode Tidak Terbatas T2/T3/T4g](#).

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 metrik. CloudWatch `CPUSurplusCreditBalance`
Kredit surplus yang dibebankan dilacak oleh metrik. CloudWatch `CPUSurplusCreditsCharged`
Untuk informasi selengkapnya, lihat [CloudWatch Metrik tambahan untuk instans performa burstable](#).

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 akan menerima kredit peluncuran karena instans tersebut mendukung mode Tidak Terbatas. Konfigurasi kredit mode Tidak Terbatas memungkinkan instans T4g, T3a dan T3 untuk menggunakan CPU sebanyak yang diperlukan untuk melonjak di atas batas dasar dan selama 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 lebih lanjut, lihat [Meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas atau Standar](#) dan [Modifikasi spesifikasi kredit dari instans performa yang dapat melonjak](#)

Anda dapat menetapkan `unlimited` sebagai opsi kredit default di tingkat akun per AWS Wilayah, per keluarga instans performa burstable, sehingga semua instance performa burstable baru di akun diluncurkan menggunakan opsi kredit default. Untuk informasi selengkapnya, lihat [Mengatur 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 lebih lanjut, lihat [Untuk mengubah spesifikasi kredit dari instans performa yang dapat melonjak](#) dan [Melihat spesifikasi kredit default](#)

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 sebuah 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 untuk diberi tahu tentang penggunaan kredit. Untuk informasi selengkapnya, lihat [Pantau kredit CPU Anda untuk instans performa yang dapat melonjak](#).

Contoh mode tidak terbatas

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.

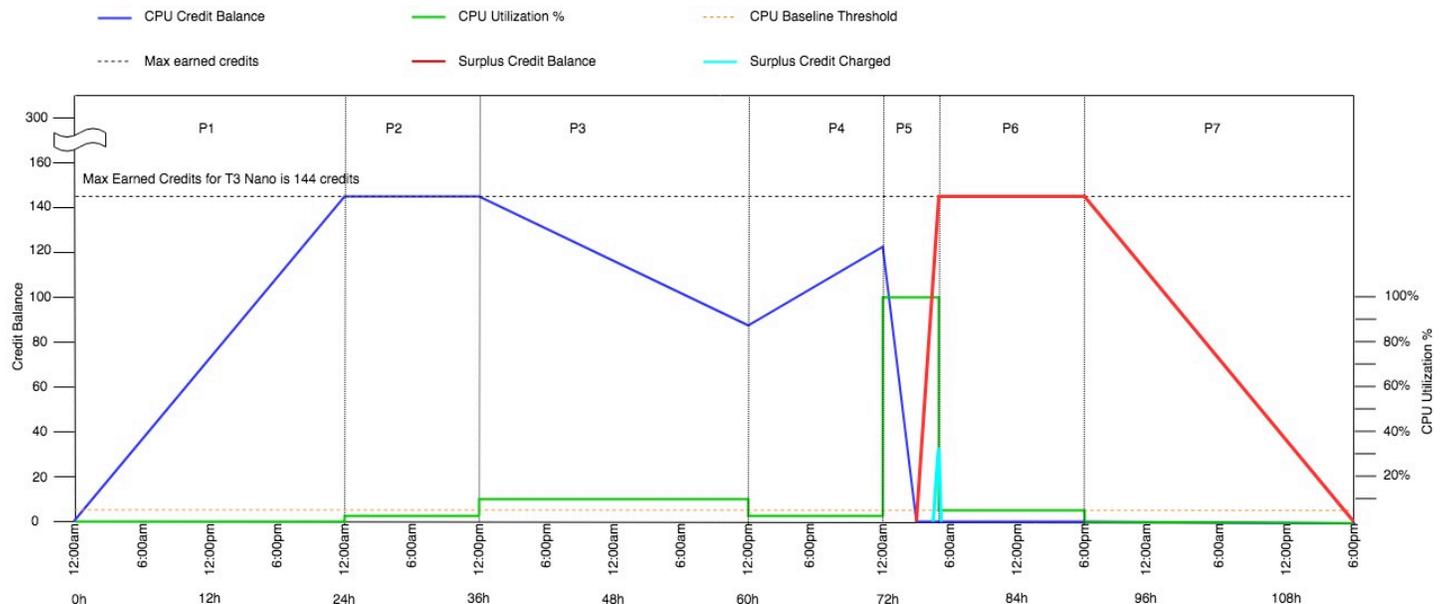
P4 - Selama 12 jam ke depan, pemanfaatan CPU menurun jadi 2,5% (di bawah acuan), yang membutuhkan penggunaan 36 kredit. Pada saat yang sama, instans tersebut mendapatkan 72 kredit. Instance mendapatkan lebih banyak kredit daripada yang dibelanjakan, dan `CPUCreditBalance` nilai meningkat menjadi 122 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 `CPU surplusCreditBalance` 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 `CPU surplusCreditBalance`. Nilai `CPU surplusCreditBalance` 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 `CPU surplusCreditBalance`.



Contoh 2: Menjelaskan penggunaan kredit dengan T2 Tidak Terbatas

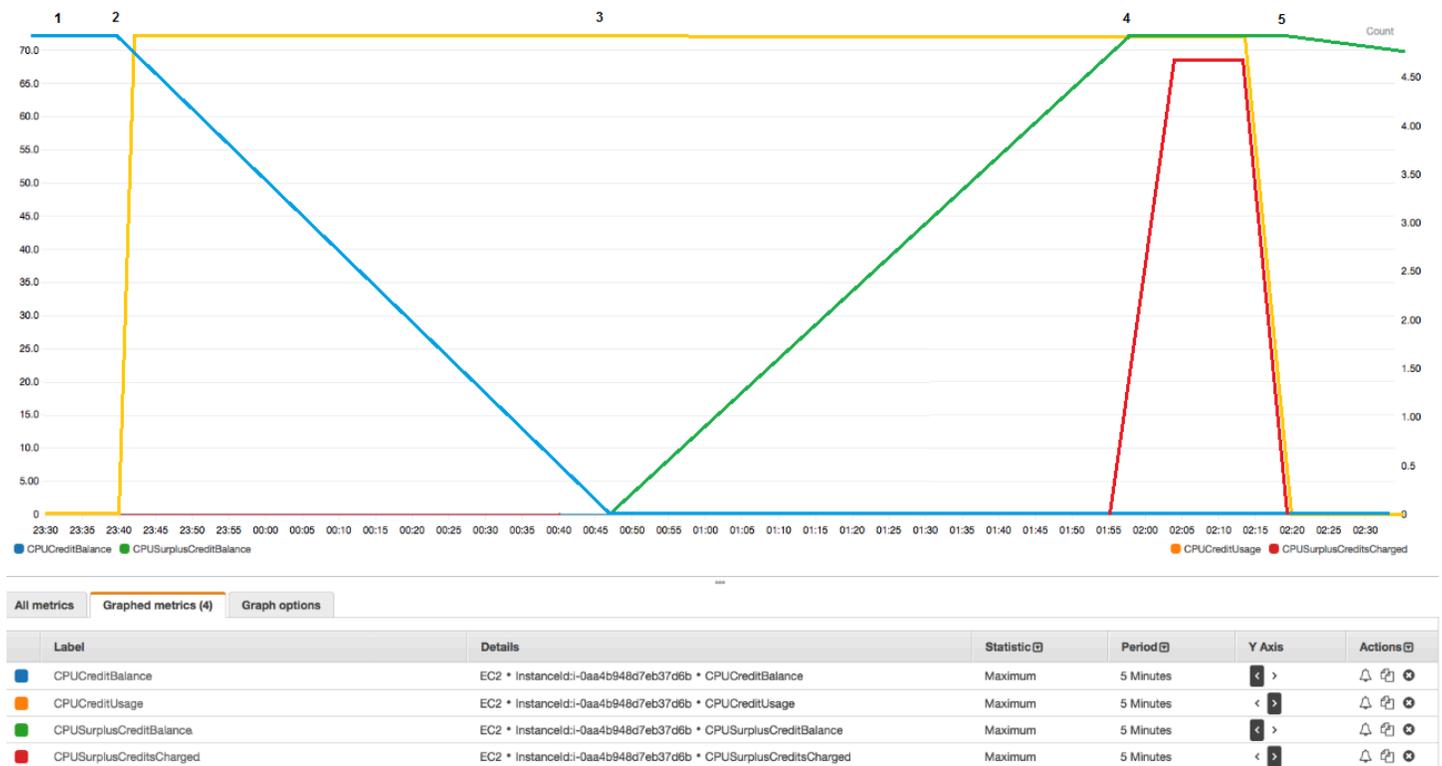
Dalam contoh ini, Anda melihat pemanfaatan CPU dari instans `t2.nano` yang diluncurkan sebagai `unlimited`, dan caranya menggunakan kredit yang diperoleh dan surplus untuk mempertahankan pemanfaatan CPU.

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 `CPU creditBalance`), 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 (instance Linux)

Kredit surplus biaya \$0.05 per VCPU-jam. Instans menggunakan sekitar 25 kredit surplus antara 01:55 dan 02:20, yang setara dengan 0,42 vCPU-jam. Biaya tambahan untuk contoh ini adalah 0,42 VCPU-jam 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 (instance Windows)

Kredit surplus biaya \$0.096 per VCPU-jam. Instans menggunakan sekitar 25 kredit surplus antara 01:55 dan 02:20, yang setara dengan 0,42 vCPU-jam. Biaya tambahan untuk contoh ini adalah 0,42 VCPU-jam 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 diakumulasi 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 utama dan definisi untuk instans performa yang dapat melonjak](#).

Daftar Isi

- [Konsep mode standar](#)
 - [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](#)
 - [Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standar](#)
 - [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

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 AWS Wilayah, per keluarga instans performa burstable, sehingga semua instance performa 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 T4g, T3a dan T3 Standar, saldo kredit CPU bertahan selama tujuh hari setelah instans berhenti dan kreditnya hilang setelah itu. 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 instans tersebut mendukung mode Tidak Terbatas. Konfigurasi kredit mode Tidak Terbatas memungkinkan instans T4g, T3a dan T3 untuk menggunakan CPU sebanyak yang diperlukan untuk melonjak di atas batas dasar dan selama 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
<code>t1.micro</code>	15	1
<code>t2.nano</code>	30	1
<code>t2.micro</code>	30	1
<code>t2.small</code>	30	1
<code>t2.medium</code>	60	2
<code>t2.large</code>	60	2
<code>t2.xlarge</code>	120	4
<code>t2.2xlarge</code>	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	<p>Instans T2 Standar memperoleh 30 kredit peluncuran per vCPU saat diluncurkan atau dimulai.</p> <p>Jika instans T2 dialihkan dari <code>unlimited</code> ke <code>standard</code>, instans ini tidak mendapatkan kredit peluncuran pada saat peralihan.</p>	<p>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.</p>
Batas perolehan kredit	<p>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.</p>	<p>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.</p>

	Kredit yang diluncurkan	Kredit yang diperoleh
Penggunaan kredit	Kredit peluncuran digunakan terlebih dahulu, sebelum kredit yang diperoleh.	Kredit yang diperoleh hanya digunakan setelah semua kredit peluncuran dihabiskan.
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 yang masih harus dibayar dilacak oleh metrik. CloudWatch `CPUCreditBalance` Untuk informasi selengkapnya, lihat `CPUCreditBalance` di [tabel CloudWatch metrik](#).

Contoh mode standar

Contoh berikut menjelaskan penggunaan kredit saat instans dikonfigurasi sebagai `standard`.

Contoh

- [Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standar](#)
- [Contoh 2: Menjelaskan penggunaan kredit dengan T2 Standar](#)

Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standar

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.

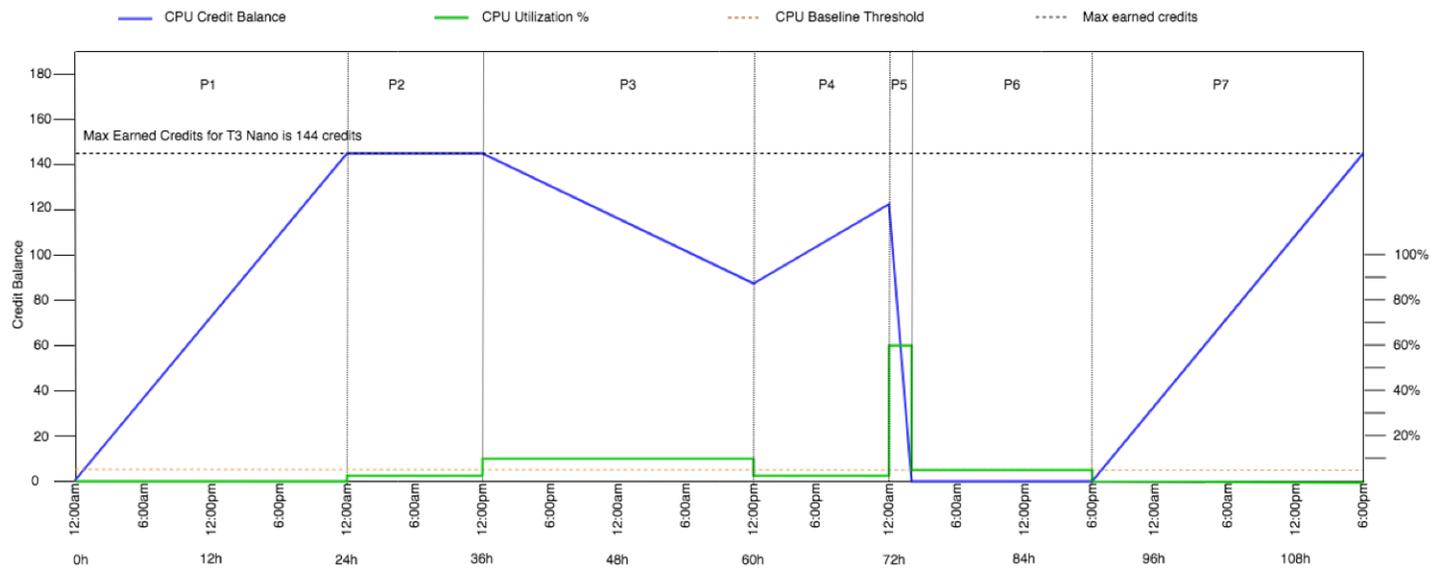
P3 - Untuk 24 jam ke depan, penggunaan CPU berada pada 7% (di atas acuan), yang membutuhkan penggunaan 57,6 kredit. Instans menggunakan lebih banyak kredit daripada yang diperolehnya, dan nilai `CPUCreditBalance` berkurang menjadi 86,4 kredit.

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, yang 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 batasan, lihat [Batas kredit peluncuran](#).

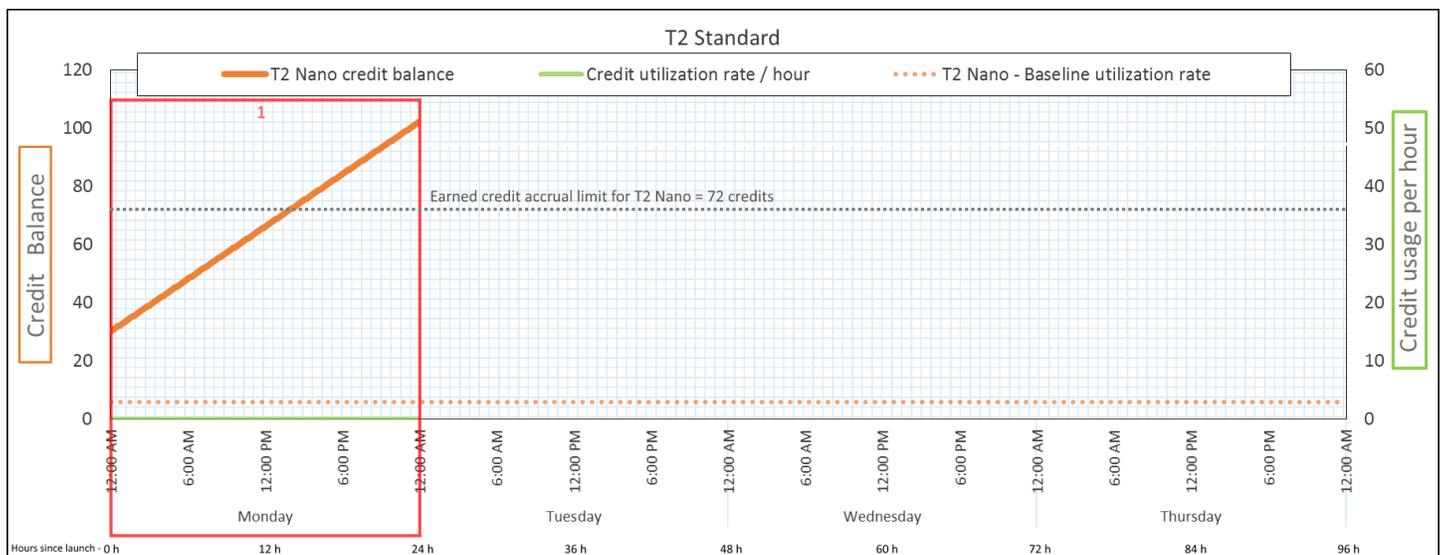
Anda dapat meluncurkan instans T2 Standar dan segera menggunakannya. Atau, Anda dapat meluncurkan instans T2 Standar dan membiarkannya idle selama beberapa hari sebelum menjalankan aplikasi di dalamnya. Digunakan atau tidaknya suatu instans akan menentukan apakah kredit akan digunakan atau diakumulasi. 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.



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)
--------------	--

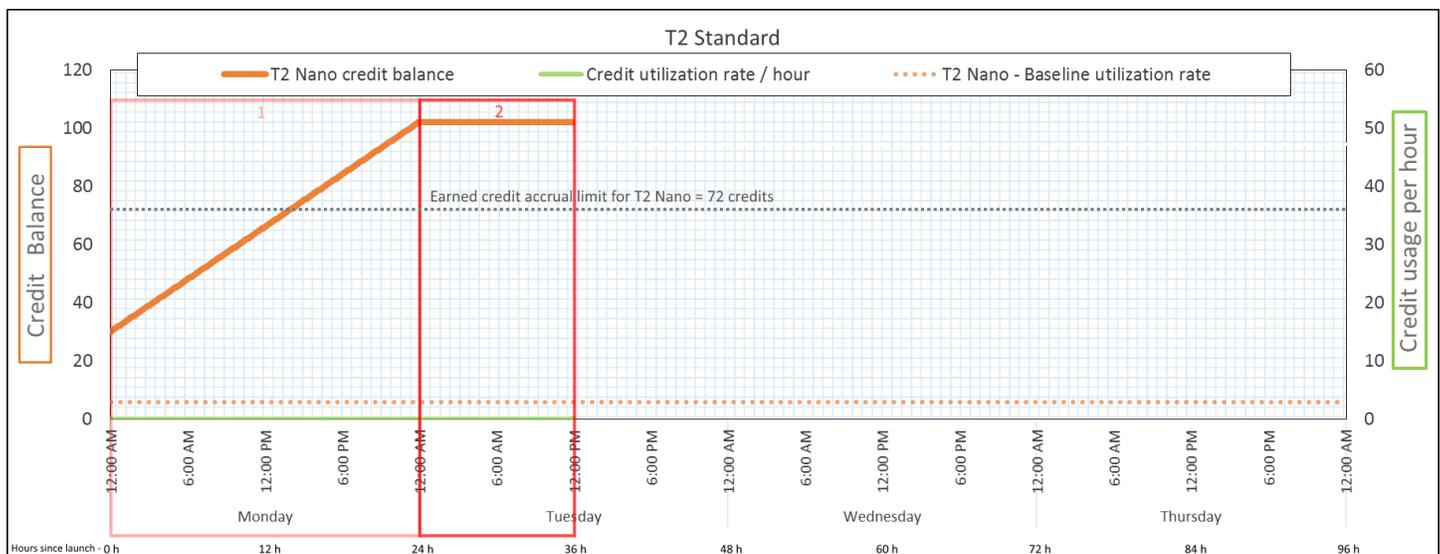
Kesimpulan

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 kredit 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.



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

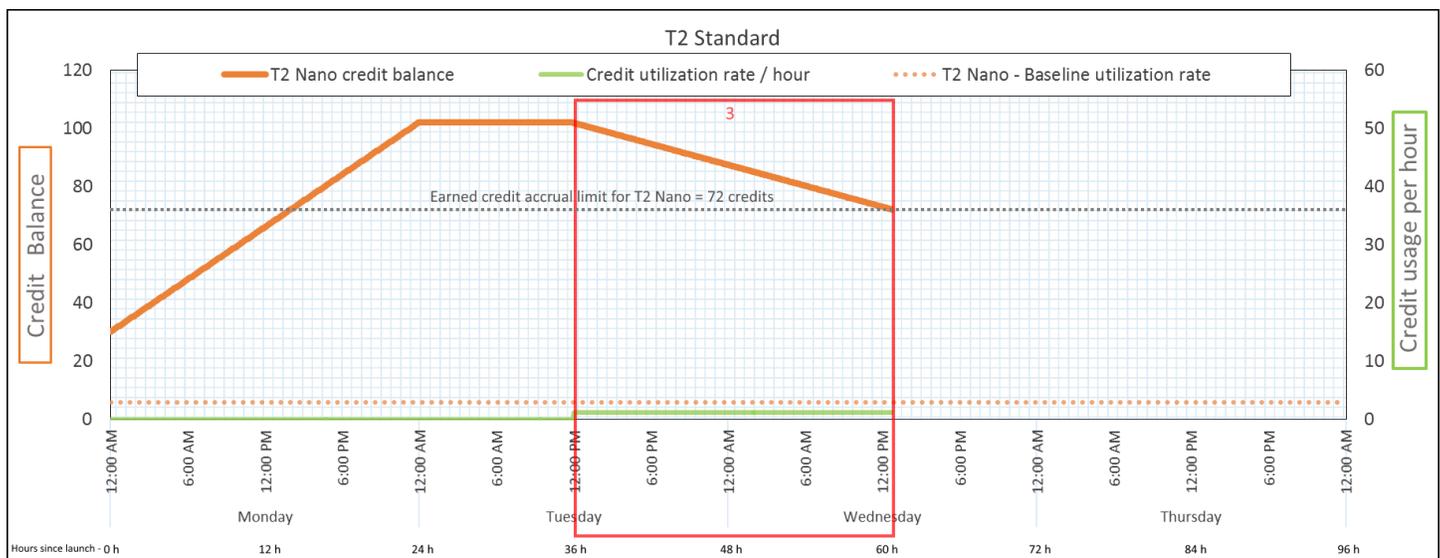
102 kredit (30 kredit peluncuran + 72 kredit yang diperoleh)—saldo tidak berubah

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.



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)

Saldo Kredit

72 kredit (30 kredit peluncuran digunakan; 72 kredit yang diperoleh tetap tidak digunakan)

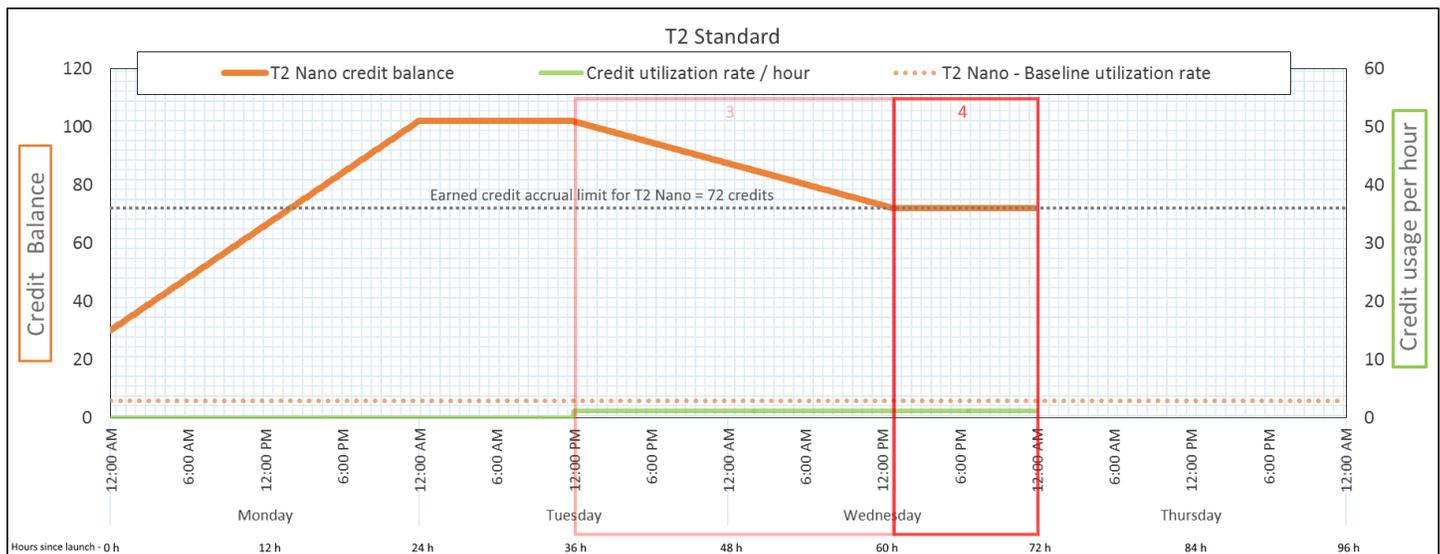
Kesimpulan

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.



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

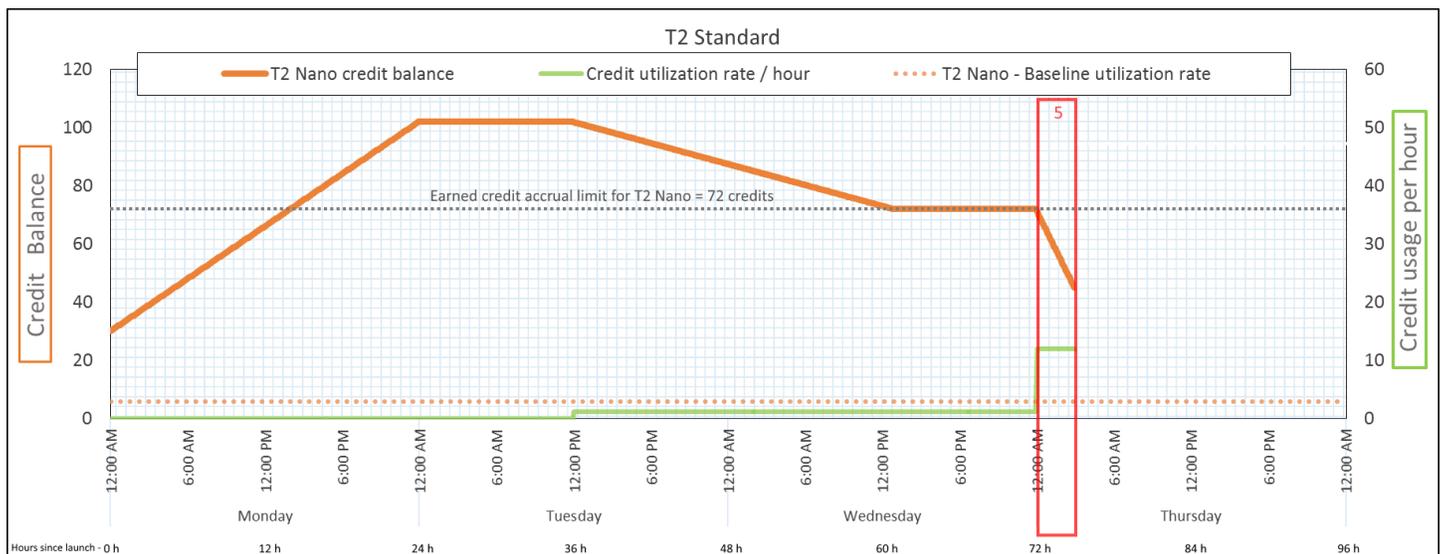
72 kredit (0 kredit peluncuran, 72 kredit yang diperoleh)—saldo berada pada batasnya

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.



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 digunakan (36) + kredit yang diperoleh (9)) —saldo menurun pada tingkat 216 kredit per 24 jam (tingkat penggunaan 288/24 + tingkat perolehan 72/24 = tingkat penurunan saldo 216/24)

Kesimpulan

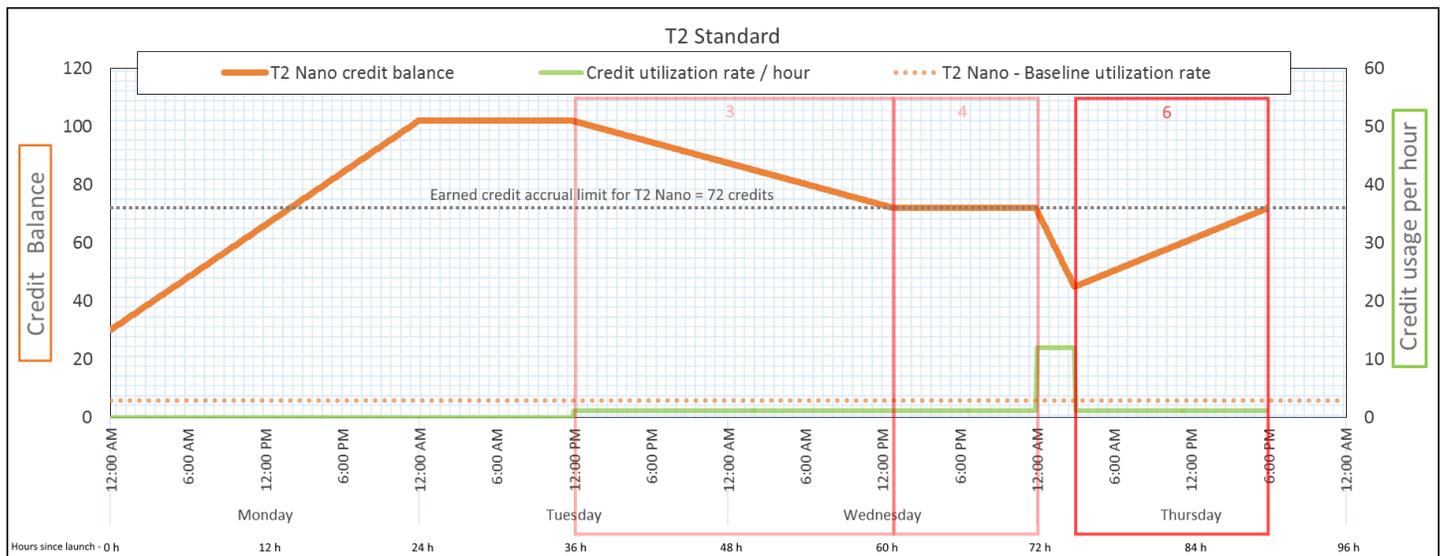
Jika sebuah instans menggunakan kredit lebih banyak daripada yang diperolehnya, saldo kreditnya menurun.

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.



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 penggunaan 28,8/24 + tingkat perolehan 72/24)

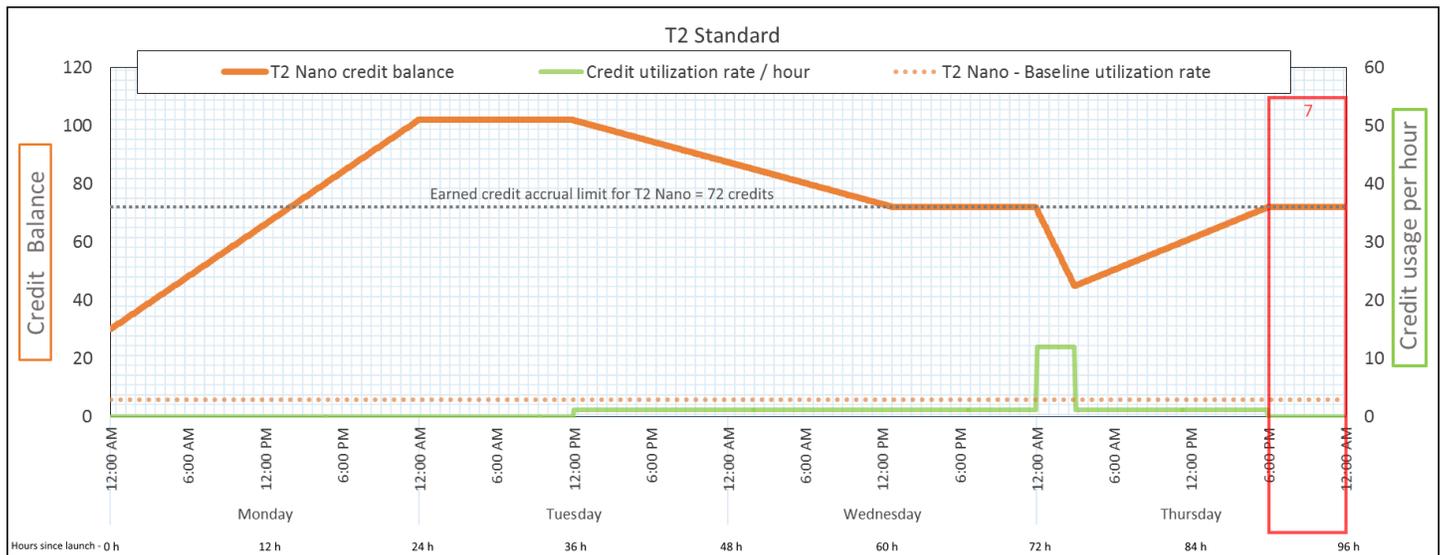
Kesimpulan

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.



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](#).

Bekerja dengan instans performa yang dapat melonjak

Langkah-langkah untuk meluncurkan, memantau, dan memodifikasi instans kinerja burstable (instans T) serupa. Perbedaan utamanya adalah spesifikasi kredit default saat diluncurkan.

Setiap keluarga instans 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.

Daftar Isi

- [Meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas atau Standar](#)
- [Menggunakan grup Auto Scaling untuk meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas](#)
- [Untuk mengubah spesifikasi kredit dari instans performa yang dapat melonjak](#)
- [Modifikasi spesifikasi kredit dari instans performa yang dapat melonjak](#)
- [Mengatur spesifikasi kredit default untuk akun](#)
- [Melihat spesifikasi kredit default](#)

Meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas atau Standar

Anda dapat meluncurkan instans T sebagai `unlimited` atau `standard` menggunakan konsol Amazon EC2, SDK, AWS alat baris perintah, atau dengan grup Auto Scaling.

Prosedur berikut menjelaskan cara menggunakan konsol EC2 atau AWS CLI Untuk informasi tentang menggunakan grup Auto Scaling, lihat. [Menggunakan grup Auto Scaling untuk meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas](#)

Console

Untuk meluncurkan instance T sebagai `Unlimited` atau `Standard`

1. Ikuti prosedur untuk [meluncurkan instans](#).
2. Pada Tipe instans, pilih tipe instans T.
3. Perluas Detail lanjutan, dan untuk Spesifikasi kredit, pilih spesifikasi kredit. Jika Anda tidak membuat pilihan, default digunakan, yaitu `standard` untuk T2, dan untuk T4G, T3a dan `unlimited` T3.

4. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

AWS CLI

Untuk meluncurkan instance T sebagai Unlimited atau Standard

Gunakan perintah [run-instances](#) untuk meluncurkan instans Anda. Tentukan spesifikasi kreditnya menggunakan parameter `--credit-specification CpuCredits=`. Spesifikasi kredit yang valid adalah `unlimited` dan `standard`

- Untuk T4G, T3a dan T3, jika Anda tidak menyertakan `--credit-specification` parameter, instance akan diluncurkan sebagai default. `unlimited`
- Untuk T2, jika Anda tidak menyertakan parameter `--credit-specification`, instans diluncurkan sebagai `standard` secara default.

```
aws ec2 run-instances \  
  --image-id ami-abc12345 \  
  --count 1 \  
  --instance-type t3.micro \  
  --key-name MyKeyPair \  
  --credit-specification "CpuCredits=unlimited"
```

Menggunakan grup Auto Scaling untuk meluncurkan instans performa yang dapat melonjak sebagai Tidak Terbatas

Ketika instans T diluncurkan atau dimulai, mereka memerlukan kredit CPU untuk pengalaman bootstrap yang baik. Jika Anda menggunakan grup Auto Scaling untuk meluncurkan instans, sebaiknya konfigurasi 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`.

Note

Mode `unlimited` tidak didukung untuk instans T3 yang diluncurkan pada Host Khusus.

Console

Untuk membuat templat peluncuran yang akan meluncurkan instans sebagai Tidak Terbatas

1. Ikuti [Buat template peluncuran menggunakan prosedur pengaturan lanjutan](#) di Panduan Pengguna Auto Scaling 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 templat peluncuran yang akan meluncurkan instans sebagai Tidak Terbatas

Gunakan perintah [create-launch-template](#) dan tentukan `unlimited` sebagai spesifikasi kredit.

- Untuk T4G, T3a dan T3, jika Anda tidak menyertakan `CreditSpecification={CpuCredits=unlimited}` nilainya, instance akan diluncurkan sebagai default. `unlimited`
- Untuk T2, jika Anda tidak menyertakan nilai `CreditSpecification={CpuCredits=unlimited}`, instans diluncurkan sebagai `standard` secara default.

```
aws ec2 create-launch-template \  
  --launch-template-name MyLaunchTemplate \  
  --version-description FirstVersion \  
  --launch-template-data  
ImageId=ami-8c1be5f6,InstanceType=t3.medium,CreditSpecification={CpuCredits=unlimited}
```

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 Wilayah yang sama dengan 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.

AWS CLI

Untuk membuat grup Auto Scaling menggunakan templat peluncuran

Gunakan perintah [create-auto-scaling-group](#) AWS CLI dan tentukan parameternya. `--launch-template`

Console

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 menambahkan templat peluncuran ke grup Auto Scaling yang ada

Gunakan perintah [update-auto-scaling-group](#) AWS CLI dan tentukan parameternya. `--launch-template`

Untuk mengubah spesifikasi kredit dari instans performa yang dapat melonjak

Anda dapat melihat spesifikasi kredit (`unlimited` atau `standard`) dari instans T yang sedang berjalan atau dihentikan.

Console

Untuk melihat spesifikasi kredit dari instance T

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans.
4. Pilih Detail dan lihat bidang Spesifikasi kredit. Nilainya adalah `unlimited` atau `standard`.

AWS CLI

Untuk menggambarkan spesifikasi kredit dari instance T

Gunakan perintah [describe-instance-credit-specifications](#). Jika Anda tidak menentukan satu atau lebih ID instans, semua instans dengan spesifikasi kredit `unlimited` dikembalikan, serta instans yang sebelumnya dikonfigurasi dengan spesifikasi kredit `unlimited`. 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
```

Contoh Output

```
{
  "InstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0",
```

```
    "CpuCredits": "unlimited"  
  }  
]  
}
```

Modifikasi spesifikasi kredit dari instans performa yang dapat melonjak

Anda dapat mengganti spesifikasi kredit dari instans T yang sedang berjalan atau berhenti 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 memodifikasi spesifikasi kredit dari instans T

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans. Untuk mengubah spesifikasi kredit untuk beberapa instans sekaligus, pilih semua instans yang berlaku.
4. Pilih Tindakan, Pengaturan instans, Ubah spesifikasi kredit. Opsi ini diaktifkan hanya jika Anda memilih instance T.
5. Untuk mengubah spesifikasi kredit menjadi `unlimited`, pilih kotak centang di sebelah ID instans. Untuk mengubah spesifikasi kredit menjadi `standard`, hapus kotak centang di samping ID instans.

AWS CLI

Untuk memodifikasi spesifikasi kredit dari instans T

Gunakan perintah [modify-instance-credit-specification](#). Tentukan instans dan spesifikasi kreditnya menggunakan parameter `--instance-credit-specification`. Spesifikasi kredit yang valid adalah `unlimited` dan `standard`

```
aws ec2 modify-instance-credit-specification \  
  --region us-east-1 \  
  --instance-id i-1234567890 \  
  --instance-credit-specification unlimited
```

```
--instance-credit-specification
"InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

Contoh Output

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i- 1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

Mengatur spesifikasi kredit default untuk akun

Setiap keluarga instans T dilengkapi dengan [spesifikasi kredit default](#). Anda dapat mengubah spesifikasi kredit default untuk setiap keluarga instans T di tingkat akun per AWS Wilayah.

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 instance AWS CLI to launch, semua instans T baru dalam peluncuran akun 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 mengatur spesifikasi kredit default di tingkat akun per Wilayah

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 kiri, pilih Dasbor EC2.
4. Dari Atribut akun, pilih Spesifikasi kredit default.
5. Pilih Kelola.

6. Untuk setiap keluarga instans, pilih Tak Terbatas atau Standard, lalu pilih Perbarui.

AWS CLI

Untuk mengatur spesifikasi kredit default untuk tingkat akun (AWS CLI)

Gunakan perintah [modify-default-credit-specification](#). Tentukan Wilayah AWS , keluarga instans, dan spesifikasi kredit default menggunakan parameter `--cpu-credits`. Spesifikasi kredit default yang valid adalah `unlimited` dan `standard`

```
aws ec2 modify-default-credit-specification \  
  --region us-east-1 \  
  --instance-family t2 \  
  --cpu-credits unlimited
```

Melihat spesifikasi kredit default

Anda dapat melihat spesifikasi kredit default dari keluarga instans T di tingkat akun per AWS Wilayah.

Console

Untuk melihat spesifikasi kredit default di tingkat akun

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 kiri, pilih Dasbor EC2.
4. Dari Atribut akun, pilih Spesifikasi kredit default.

AWS CLI

Untuk melihat spesifikasi kredit default di tingkat akun

Gunakan perintah [get-default-credit-specification](#). Tentukan Wilayah AWS dan keluarga instans.

```
aws ec2 get-default-credit-specification --region us-east-1 --instance-family t2
```

Pantau kredit CPU Anda untuk instans performa yang dapat melonjak

EC2 mengirimkan metrik ke Amazon. CloudWatch Anda dapat melihat metrik kredit CPU di metrik Amazon EC2 per instans konsol atau menggunakan AWS CLI metrik untuk mencantumkan metrik untuk setiap instans. CloudWatch Untuk informasi lebih lanjut, lihat [Membuat daftar metrik menggunakan konsol](#) dan [Daftar metrik menggunakan AWS CLI](#)

Daftar Isi

- [CloudWatch Metrik tambahan untuk instans performa burstable](#)
- [Menghitung penggunaan kredit CPU](#)

CloudWatch Metrik tambahan untuk instans performa burstable

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 instance kinerja burstable. Untuk informasi selengkapnya, lihat [Buat daftar CloudWatch metrik yang tersedia untuk instans Anda](#).

Metrik	Deskripsi
<code>CPUCreditUsage</code>	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

Metrik	Deskripsi
	<p>an 50% selama dua menit atau dua vCPU yang berjalan dengan pemanfaatan 25% selama dua menit).</p> <p>Metrik kredit CPU tersedia pada frekuensi lima menit saja. Jika Anda menentukan periode lebih dari lima menit, gunakan statistik, Sumbukan statistik.Average</p> <p>Unit: Kredit (vCPU-menit)</p>
CPUCreditBalance	<p>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.</p> <p>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.</p> <p>Kredit dalam CPUCreditBalance tersedia untuk instans untuk digunakan hingga melonjak melebihi pemanfaatan CPU acuan.</p> <p>Saat sebuah instans berjalan, kredit di CPUCreditBalance tidak kedaluarsa. Ketika instans T4G, T3a atau T3 berhenti, CPUCreditBalance nilainya 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.</p> <p>Metrik kredit CPU tersedia pada frekuensi lima menit saja.</p> <p>Unit: Kredit (vCPU-menit)</p>

Metrik	Deskripsi
CPUSurplusCreditBalance	<p>Jumlah kredit surplus yang telah digunakan oleh instans <code>unlimited</code> saat nilai <code>CPUCreditBalance</code> miliknya adalah nol.</p> <p>Nilai <code>CPUSurplusCreditBalance</code> 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 dikenakan biaya tambahan.</p> <p>Unit: Kredit (vCPU-menit)</p>
CPUSurplusCreditsCharged	<p>Jumlah kredit surplus yang digunakan yang tidak dibayarkan oleh kredit CPU yang diperoleh, dikenakan biaya tambahan.</p> <p>Kredit surplus yang digunakan akan dikenai biaya jika salah satu dari hal berikut terjadi:</p> <ul style="list-style-type: none"> • 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. • Instans dihentikan atau diakhiri. • Instans dialihkan dari <code>unlimited</code> ke <code>standard</code>. <p>Unit: Kredit (vCPU-menit)</p>

Menghitung penggunaan kredit CPU

Penggunaan kredit CPU dari instance 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

$$\text{CPUCreditBalance} = \text{prior CPUCreditBalance} + [\text{Credits earned per hour} * (5/60) - \text{CPUCreditUsage}]$$

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`— Merupakan interval lima menit antara publikasi CloudWatch 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

$$\text{CPUCreditBalance} = 2 + [0.5 - 1] = 1.5$$

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 menggunakan kredit surplus. Saat menggunakan saldo kredit CPU yang diakumulasi, instans dapat menggunakan kredit surplus untuk melonjatkan 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 `CPUCreditBalance` dan `CPUSurplusCreditBalance` CloudWatch metrik.

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 `CPU surplus credits charged`.

Example

```
CPU surplus credits charged = max [-Adjusted balance - max earned credit balance, 0]
```

Akhirnya, saat instans berakhir, semua kredit surplus yang dilacak oleh `CPU surplus credit balance` dikenakan tagihan. Jika instans dialihkan dari `unlimited` ke `standard`, setiap `CPU surplus credit balance` yang tersisa juga dikenakan biaya.

Akselerasi kinerja dengan instans GPU

Instans berbasis GPU menyediakan akses ke GPU NVIDIA dengan ribuan core 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 instance berbasis GPU, Anda harus menginstal driver yang sesuai, sebagai berikut:

- Untuk menginstal driver NVIDIA pada instance dengan GPU NVIDIA yang terpasang, seperti instance P3 atau G4dn, lihat. [Menginstal driver NVIDIA](#)
- Untuk menginstal driver AMD pada instance dengan GPU AMD yang terpasang, seperti instance G4ad, lihat. [Menginstal driver AMD](#)

Daftar Isi

- [Aktifkan Aplikasi Virtual NVIDIA GRID di instans berbasis GPU Amazon EC2](#)
- [Optimalkan pengaturan GPU pada instans Amazon EC2](#)
- [Mengatur tampilan 4K Ganda pada instans Linux G4ad](#)
- [Memulai instans P5 untuk Linux](#)

Aktifkan Aplikasi Virtual NVIDIA GRID di instans berbasis GPU Amazon EC2

Untuk mengaktifkan Aplikasi Virtual GRID pada instance berbasis GPU yang memiliki GPU NVIDIA (NVIDIA GRID Virtual Workstation diaktifkan secara default), Anda harus menentukan jenis produk untuk driver, sebagai berikut.

Aktifkan Aplikasi Virtual GRID pada instance Linux

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
```

Aktifkan Aplikasi Virtual GRID pada instance Windows

Aktifkan Aplikasi Virtual GRID pada instance Windows

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, masukkan `FeatureType` dan ketik `Enter`.
5. Buka menu konteks (klik kanan) `FeatureType` dan pilih `Ubah`.
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 autoboot 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. Konfigurasi pengaturan GPU agar menjadi persisten. Perintah ini memerlukan waktu beberapa menit untuk dijalankan.

```
[ec2-user ~]$ sudo nvidia-persistenced
```

2. [Hanya instans G3, dan P2] Nonaktifkan fitur autoboot 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 dan Gr6:

```
[ec2-user ~]$ sudo nvidia-smi -ac 6251,2040
```

- Instans P2:

```
[ec2-user ~]$ sudo nvidia-smi -ac 2505,875
```

- 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
```

Optimalkan pengaturan GPU di Windows

1. Buka PowerShell jendela dan arahkan ke folder instalasi NVIDIA.

```
cd "C:\Windows\System32\DriverStore\FileRepository\nv_dispswi.inf_*\"
```

2. [Hanya instans G3, dan P2] Nonaktifkan fitur autoboot untuk semua GPU pada instance.

```
.\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:

```
.\nvidia-smi -ac "2505,1177"
```

- Instans G4dn:

```
.\nvidia-smi -ac "5001,1590"
```

- Instans G5:

```
.\nvidia-smi -ac "6250,1710"
```

- Instans G6 dan Gr6:

```
.\nvidia-smi -ac "6251,2040"
```

- Instans P2:

```
.\nvidia-smi -ac "2505,875"
```

- Instans P3 dan P3dn:

```
.\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
```

Mengatur tampilan 4K Ganda pada instans Linux G4ad

Luncurkan instans G4ad

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, lihat [Instal driver AMD di instans Amazon 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.



Memulai instans P5 untuk Linux

Instans P5 menyediakan 8 GPU NVIDIA H100 dengan memori GPU bandwidth tinggi 640 GB. Instans ini memiliki prosesor AMD EPYC generasi ke-3 dan menyediakan memori sistem 2 TB, penyimpanan instans NVMe lokal 30 TB, bandwidth jaringan agregat 3.200 Gbps, dan dukungan GPUDirect RDMA. Instans P5 juga mendukung teknologi Amazon UltraCluster EC2, yang memberikan latensi lebih rendah dan peningkatan kinerja jaringan menggunakan EFA.

Tabel berikut memberikan ringkasan spesifikasi p5.48xlarge.

vCPU	Memori sistem	GPU	Memo GPU	Bandwidth jaringan	GPUDirect RDMA	GPU peer to peer	Penyimpanan instans
192	2 TiB	8 GPU NVIDIA H100	640 GB HBM3	3200 Gbps dengan eFAV2	Didukung	900 GB/s NVSw	Volume SSD NVMe 8 x 3.800 GB

Konfigurasi perangkat lunak

Cara termudah untuk memulai dengan instans P5 adalah dengan meluncurkan instans menggunakan AWS Deep Learning AMI yang telah dikonfigurasi sebelumnya dengan semua perangkat lunak yang diperlukan. Untuk yang terbaru AWS Deep Learning AMI untuk digunakan dengan instance P5, lihat [AMI GPU AWS Deep Learning Base \(Ubuntu 20.04\)](#).

Jika Anda perlu membangun AMI kustom untuk digunakan dengan instans P5, sebaiknya instal versi perangkat lunak minimum berikut ini:

- Driver NVIDIA 535.54.03 atau setelahnya
- CUDA 12.1 atau setelahnya
- NVIDIA GDRCopy 2.3 atau versi terbaru
- Penginstal EFA 1.24.1 atau setelahnya
- NCCL 2.18.3 atau setelahnya
- aws-ofi-nccl plugin 1.7.2-aws atau yang lebih baru

Kami juga menyarankan agar Anda mengonfigurasi instans agar tidak menggunakan status C yang lebih dalam. Untuk informasi selengkapnya, lihat [Kinerja tinggi dan latensi rendah dengan membatasi status C yang lebih](#) dalam di Panduan Pengguna Amazon Linux 2. GPU AMI AWS Deep Learning Base terbaru telah dikonfigurasi sebelumnya untuk tidak menggunakan status C yang lebih dalam.

Rekomendasi khusus Ubuntu 20.04

Rekomendasi berikut untuk Ubuntu 20.04 membantu mencegah penamaan antarmuka yang tidak terduga saat boot:

- Pastikan Anda menjalankan `systemd 245.4-4ubuntu3.19` atau versi terbaru dengan perintah berikut:

```
systemd --version
```

- Pastikan Anda telah mengonfigurasi GRUB:
 - Buka file konfigurasi `/etc/default/grub` di editor teks.
 - Edit entri `GRUB_CMDLINE_LINUX_DEFAULT` untuk menyertakan `net.naming-scheme=v247`.
 - Lakukan boot ulang instans Anda dengan menjalankan `sudo update-grub`.

Konfigurasi jaringan dan EFA

Instans P5 menghasilkan bandwidth jaringan 3200 Gbps dengan menggunakan beberapa antarmuka EFA. Instans P5 mendukung 32 kartu jaringan. Kami menyarankan Anda menentukan antarmuka jaringan EFA tunggal per kartu jaringan. Untuk mengonfigurasi antarmuka ini saat peluncuran, kami merekomendasikan pengaturan berikut:

- Untuk antarmuka jaringan 0, tentukan indeks perangkat 0
- Untuk antarmuka jaringan 1 melalui 31, tentukan indeks perangkat 1

Untuk informasi selengkapnya tentang cara mengonfigurasi instans P5 Anda untuk EFA, lihat [Memulai instans P5 dan EFA](#).

Instans Mac Amazon EC2

Instans Amazon EC2 Mac secara native mendukung sistem operasi macOS.

- Instans EC2 x86 (`mac1.meta1`) dibangun di atas perangkat keras mini Mac 2018 yang didukung oleh Prosesor Intel Core i7 generasi kedelapan (Coffee Lake) 3,2 GHz.
- Instans EC2 M1 Mac (`mac2.meta1`) dibangun di atas perangkat keras mini Mac 2020 yang didukung oleh prosesor Apple silicon M1.
- Instans EC2 M2 Mac (`mac2-m2.meta1`) dibangun di atas perangkat keras mini Mac 2023 yang didukung oleh prosesor Apple silicon M2.
- Instans EC2 M2 Pro Mac (`mac2-m2pro.meta1`) dibangun di atas perangkat keras mini Mac 2023 yang didukung oleh prosesor Apple silicon M2 Pro.

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.

Daftar Isi

- [Pertimbangan](#)

- [Kesiapan instans](#)
- [AMI macOS EC2](#)
- [EC2 macOS Init](#)
- [Monitor Sistem Amazon EC2 untuk macOS](#)
- [Sumber daya terkait](#)
- [Luncurkan instans Mac](#)
- [Terhubung ke instans Mac Anda](#)
- [Perbarui sistem operasi dan perangkat lunak pada instance Mac](#)
- [Meningkatkan ukuran volume EBS di instans Mac Anda](#)
- [Menghentikan dan mengakhiri instans Mac Anda](#)
- [Temukan versi macOS yang didukung untuk Host Khusus Amazon EC2 Mac](#)
- [Berlangganan notifikasi AMI macOS](#)
- [Catatan rilis Amazon EC2 macOS AMI](#)

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 jenis Wilayah AWS. Untuk daftar ketersediaan instans Mac di masing-masing instans Wilayah AWS, lihat [jenis instans Amazon EC2 menurut Wilayah](#).
- Instans Mac hanya tersedia sebagai Instans Sesuai Permintaan. Instans tersebut tidak tersedia sebagai Instans Spot atau Instans Terpesan. Anda dapat menghemat uang pada instans Mac dengan membeli [Savings Plans](#).
- 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)
- EBSshotplug didukung.
- AWS tidak mengelola atau mendukung SSD internal pada perangkat keras Apple. Kami sangat menyarankan agar Anda menggunakan volume Amazon EBS sebagai gantinya. EBSvolume memberikan manfaat elastisitas, ketersediaan, dan daya tahan yang sama pada instans Mac seperti halnya pada instans EC2 lainnya.
- Sebaiknya gunakan General Purpose SSD (gp2dangp3) dan Provisioned IOPS SSD (io1daniio2) dengan instans Mac untuk kinerja yang optimal. 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 [Perbarui sistem operasi dan perangkat lunak pada instance Mac.](#)
- Ketika Anda menghentikan atau mengakhiri instans Mac, alur kerja scrubbing dilakukan pada Host Khusus. Untuk informasi selengkapnya, lihat [Menghentikan dan mengakhiri instans Mac Anda.](#)

Warning

Jangan gunakan FileVault. Mengaktifkan FileVault akan mengakibatkan host gagal 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 pada server yang meng-host instans, memastikan keamanan keduanya data-at-rest dan data-in-transit antara instans dan penyimpanan EBS terlampirnya. 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 terhubung ke instans tersebut. Untuk AWS AMI yang dijual dengan instans Mac x86 atau instans Apple silicon Mac, 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 `describe-instance-status` API untuk mengetahui kapan instance 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-0123456789example \
  --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 mencakup paket yang memungkinkan integrasi AWS yang mudah, seperti alat konfigurasi peluncuran dan pustaka serta alat AWS populer.

Untuk informasi selengkapnya tentang AMI EC2 macOS, lihat [Catatan rilis Amazon EC2 macOS AMI](#)

AWS menyediakan AMI EC2 macOS yang diperbarui secara teratur yang mencakup pembaruan paket yang dimiliki oleh AWS dan versi macOS terbaru yang telah diuji 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 macOS AMI, lihat [Berlangganan notifikasi AMI macOS](#)

EC2 macOS Init

EC2macOS Init digunakan untuk menginisialisasi instance EC2 Mac saat diluncurkan. 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

Monitor Sistem Amazon EC2 untuk macOS menyediakan metrik pemanfaatan CPU ke Amazon CloudWatch. Ini mengirimkan metrik ini ke CloudWatch lebih dari 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

Monitor Sistem Amazon EC2 untuk macOS saat ini tidak didukung pada instans Apple silikon Mac.

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

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 AWS Management Console atau file 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 instans, pilih `mac1`, `mac2`, `mac2-m2`, atau `mac2-m2pro`. Jika keluarga instans tidak muncul dalam daftar, berarti tidak didukung di Wilayah yang saat ini dipilih.
 - ii. Untuk Tipe instans, pilih `mac1.metal`, `mac2.metal`, atau `mac2-m2.metal`, atau `mac2-m2pro.metal` berdasarkan keluarga instans 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. Pada Tipe instans, pilih tipe instans yang sesuai (`mac1.metal`, `mac2.metal`, `mac2-m2.metal`, atau `mac2-m2pro.metal`).
 - 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 instance Mac menggunakan AWS CLI

Mengalokasikan Host Khusus

Gunakan perintah [allocate-hosts](#) berikut untuk mengalokasikan Host Khusus untuk instans Mac Anda, ganti `instance-type` dengan salah satu `mac1.metal`, `mac2.metal`, `mac2-m2.metal`, atau `mac2-m2pro.metal`, serta `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-instances](#) berikut untuk meluncurkan instans Mac, sekali lagi dengan mengganti `instance-type` dengan salah satu `mac1.metal`, `mac2.metal`, `mac2-m2.metal`, atau `mac2-m2pro.metal`, serta `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"
    }
  ]
}
```

Terhubung ke instans Mac Anda

Anda dapat terhubung ke instans Mac Anda menggunakan SSH atau antarmuka pengguna grafis.

Hubungkan ke instans Anda menggunakan SSH

Important

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 banyak sesi hingga batas “Sesi Maks” di file `sshd_config`.

Instans Mac Amazon EC2 tidak mengizinkan SSH root jarak jauh secara default. Autentikasi kata sandi dinonaktifkan untuk mencegah serangan kata sandi brute-force. Akun `ec2-user` dikonfigurasi untuk masuk dari jarak jauh menggunakan SSH. Akun `ec2-user` juga memiliki hak istimewa `sudo`. 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 menghubungkan ke instans 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
```

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 seterusnya 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, dengan menggunakan `ec2-user` sebagai nama pengguna dan sandi yang Anda buat untuk akun `ec2-user`.

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"
```

Sebagai contoh:

```
RES="2560x1600"
displayplacer "id:69784AF1-CD7D-B79B-E5D4-60D937407F68 res:${RES} scaling:off
origin:(0,0) degree:0"
```

Perbarui sistem operasi dan perangkat lunak pada instance Mac

Warning

Pemasangan versi beta atau pratinjau macOS hanya tersedia di instans Amazon EC2 M1 Mac. 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.

Langkah-langkah untuk memperbarui perangkat lunak pada instans Mac x86 dan instans Apple silicon Mac.

- [Perbarui perangkat lunak pada instans Mac x86](#)
- [Perbarui perangkat lunak pada instans Mac Apple silicon](#)

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 digunakan AWS Systems Manager untuk meluncurkan pembaruan sistem operasi yang telah disetujui sebelumnya pada instance 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-paket ini di instance 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
```

Perbarui perangkat lunak pada instans Mac Apple silicon

Pertimbangan

Driver Adaptor Jaringan Elastis (ENA)

Karena pembaruan dalam konfigurasi driver jaringan, driver ENA versi 1.0.2 tidak kompatibel dengan macOS 13.3 atau yang lebih tinggi. 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. Unduh aplikasi ENA ke dalam file Applications menggunakan perintah berikut.

```
[ec2-user ~]$ brew install amazon-ena-ethernet-dext
```

Tip pemecahan masalah

Jika Anda menerima peringatan No available formula with the name amazon-ena-ethernet-dext, jalankan perintah berikut.

```
[ec2-user ~]$ brew update
```

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 instans 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.

Perbarui perangkat lunak pada instans Mac Apple silicon

Pada instans Mac Apple silicon, Anda harus menyelesaikan beberapa langkah untuk melakukan pembaruan sistem operasi di tempat. Pertama, akses disk internal instans menggunakan GUI dengan klien VNC (Komputasi Jaringan Virtual). Prosedur ini menggunakan MacOS Screen Sharing, klien VNC bawaan. Kemudian, delegasikan kepemilikan kepada pengguna administratif (`ec2-user`) dengan masuk seperti `aws-managed-user` pada volume Amazon EBS.

Saat Anda mengerjakan prosedur ini, Anda membuat dua kata sandi. Satu kata sandi adalah untuk pengguna administratif (`ec2-user`) dan kata sandi lainnya adalah untuk pengguna administratif khusus (`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 [Hubungkan 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.

i 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 harus dihentikan dan dimulai ulang sehingga host khusus dapat diperbarui. Untuk informasi selengkapnya, lihat [Menghentikan dan mengakhiri instans Mac 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 menyimpannya 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 menyimpannya 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 Disk Startup) dan pilih Restart.

 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.

- Masukkan `ec2-user` kata sandi yang Anda buat di [Langkah 3](#) dari prosedur sebelumnya, lalu pilih Lanjutkan.
- Masukkan kata sandi untuk pengguna administratif khusus (`aws-managed-user`) saat diminta.
- Dari komputer lokal Anda, di Terminal, hubungkan kembali ke instans Anda menggunakan SSH dengan nama pengguna `ec2-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
```

- Untuk melakukan pembaruan perangkat lunak, gunakan perintah di bawah [Perbarui perangkat lunak pada instans Mac x86](#).

Meningkatkan ukuran volume EBS di instans Mac Anda

Anda dapat meningkatkan ukuran volume Amazon EBS pada instans Mac Anda. Untuk informasi selengkapnya, lihat [Volume Elastis Amazon EBS](#) 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

- 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 dan mengakhiri instans Mac 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 instance Mac memberikan keamanan dan privasi data yang sama seperti 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. Pada instans Apple silicon Mac, alur kerja scrubbing mungkin membutuhkan waktu hingga 110 menit untuk diselesaikan. Selain itu, pada instans Mac x86, jika firmware perangkat perlu diperbarui, maka alur kerja scrubbing mungkin membutuhkan waktu hingga 3 jam untuk menyelesaikannya.

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 Contoh.
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.

Temukan versi macOS yang didukung untuk Host Khusus Amazon EC2 Mac

Anda dapat melihat versi macOS terbaru yang didukung oleh Host Khusus Amazon EC2 Mac Anda. Dengan fungsi ini, Anda dapat memvalidasi apakah Host Khusus dapat mendukung peluncuran instans 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 Mac Dedicated Host yang dialokasikan tetap menganggur untuk jangka waktu yang lama atau jika memiliki instance yang berjalan lama di dalamnya.

Untuk memastikan dukungan untuk versi macOS terbaru, Anda dapat menghentikan atau menghentikan instans pada Host Khusus Mac yang dialokasikan. Ini memicu alur kerja scrubbing 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 instance yang sedang berjalan.

Untuk informasi selengkapnya tentang alur kerja scrubbing, lihat [Menghentikan dan mengakhiri instans Mac Anda](#)

Untuk informasi selengkapnya tentang meluncurkan instance Mac, lihat [Luncurkan instans Mac](#).

Anda dapat melihat informasi tentang versi macOS terbaru yang didukung pada 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 EC2 macOS AMI, lihat. [Catatan rilis Amazon EC2 macOS AMI](#)

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 (Pemberitahuan 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.

Catatan rilis Amazon EC2 macOS AMI

Informasi berikut memberikan detail 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 macOS AMI, lihat [Berlangganan notifikasi AMI macOS](#)

Paket default disertakan dalam AMI macOS Amazon EC2

Tabel berikut menjelaskan paket yang disertakan secara default dalam AMI macOS EC2.

Paket	Catatan rilis
EC2macOS Init	https://github.com/aws/ec2-macos-init/tags
EC2macOS Utils	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
Homebrew	https://github.com/Homebrew/brew/releases
EC2 Instance Connect	https://github.com/aws/aws-ec2-instance-connect-config/releases

Paket	Catatan rilis
Safari	https://developer.apple.com/documentation/safari-release-notes

Pembaruan macOS AMI Amazon EC2

Tabel berikut menjelaskan perubahan yang disertakan dalam rilis macOS AMI EC2. Perhatikan bahwa beberapa perubahan berlaku untuk semua AMI macOS EC2, sedangkan yang lain hanya berlaku untuk sebagian dari AMI ini.

Pembaruan macOS AMI EC2

Rilis	Perubahan
2024.06.07	<p>Semua AMI</p> <ul style="list-style-type: none"> Diperbarui Homebrew ke 4.3.1-1 Diperbarui <code>aws-cli</code> ke 2.15.56 Diperbarui <code>amazon-ssm-agent</code> ke 3.3.380.0-1 <p>Rilis macOS Sonoma 14.5 (semua instance Mac)</p> <ul style="list-style-type: none"> Konten keamanan macOS Sonoma 14.5 <p>Rilis macOS Ventura 13.6.7 (semua instance Mac)</p> <ul style="list-style-type: none"> Konten keamanan macOS Ventura 13.6.7 Diperbarui Safari ke 17.5 <ul style="list-style-type: none"> Konten keamanan Safari 17.5 <p>Rilis macOS Monterey 12.7.5 (semua instance Mac)</p> <ul style="list-style-type: none"> Konten keamanan macOS Monterey 12.7.5 Diperbarui Safari ke 17.5

Rilis	Perubahan
	<ul style="list-style-type: none"> • Konten keamanan Safari 17.5
2024.04.12	<p>Semua AMI</p> <ul style="list-style-type: none"> • Diperbarui Homebrew ke 4.2.16-1 • Diperbarui aws -cli ke 2.15.36 <p>Rilis macOS Sonoma 14.4.1 (semua instance Mac)</p> <ul style="list-style-type: none"> • Konten keamanan macOS Sonoma 14.4.1 <p>Rilis macOS Ventura 13.6.6 (semua instance Mac)</p> <ul style="list-style-type: none"> • Konten keamanan macOS Ventura 13.6.6 • Diperbarui Safari ke 17.4.1 <ul style="list-style-type: none"> • Konten keamanan Safari 17.4.1 <p>Untuk macOS Monterey (semua instance Mac)</p> <ul style="list-style-type: none"> • Diperbarui Safari ke 17.4.1 <ul style="list-style-type: none"> • Konten keamanan Safari 17.4.1

Instans yang dioptimalkan Amazon EBS

Instans yang dioptimalkan Amazon EBS menggunakan tumpukan konfigurasi yang dioptimalkan dan memberikan tambahan, kapasitas khusus untuk I/O Amazon EBS. Optimisasi ini memberikan performa terbaik untuk volume EBS Anda dengan meminimalkan pendapat antara I/O Amazon EBS dan lalu lintas lain dari instans Anda.

Instans yang dioptimalkan EBS memberikan bandwidth khusus untuk Amazon EBS. Jika dipasang ke instans yang dioptimalkan EBS, volume SSD Tujuan Umum (gp2 dan gp3) dirancang untuk memberikan setidaknya 90% performa IOPS yang tersedia selama 99% waktu di tahun tertentu, dan volume SSD IOPS yang Tersedia (io1 dan io2) dirancang untuk memberikan setidaknya 90% dari performa IOPS yang tersedia selama 99,9% waktu di tahun tertentu. HDD

Throughput yang Dioptimalkan (st1) dan Cold HDD (sc1) memberikan setidaknya 90% performa throughput yang diharapkan 99% dari waktu pada tahun tertentu. Periode yang tidak sesuai didistribusikan kurang lebih secara seragam, yang menargetkan 99% dari total throughput yang diharapkan setiap jam. Untuk informasi selengkapnya, lihat [Jenis volume Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Important

Performa EBS instans dibatasi oleh batas performa instans, atau performa agregat dari volume terlampirnya, mana 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`, instans harus memiliki setidaknya 5 volume gp3 yang disediakan dengan 16,000 IOPS masing-masing (5 volume x 16,000 IOPS = 80,000 IOPS).

Daftar Isi

- [Tipe instans yang didukung](#)
- [Dapatkan performa maksimum](#)
- [Lihat tipe instans yang mendukung optimisasi EBS](#)
- [Aktifkan optimisasi EBS saat peluncuran](#)
- [Aktifkan optimisasi EBS untuk instans yang sudah ada](#)

Tipe instans yang didukung

Tabel berikut menunjukkan tipe instans yang mendukung optimisasi EBS. Termasuk di dalamnya bandwidth khusus untuk Amazon EBS, throughput agregat maksimum umum yang dapat dicapai pada koneksi tersebut dengan beban kerja baca streaming dan ukuran I/O 128 KiB, serta IOPS maksimum yang dapat didukung instans jika Anda menggunakan ukuran I/O 16 KiB.

Pilih instans yang dioptimalkan EBS yang memberikan hasil throughput Amazon EBS yang lebih khusus daripada kebutuhan aplikasi Anda; jika tidak, koneksi antara Amazon EBS dan Amazon EC2 dapat menghambat performa.

Daftar Isi

- [EBS dioptimalkan secara standar](#)

- [Optimisasi EBS didukung](#)

EBS dioptimalkan secara standar

Tabel berikut mencantumkan tipe instans yang mendukung optimisasi EBS dan optimisasi EBS diaktifkan secara default. Tidak perlu mengaktifkan optimisasi EBS dan tidak akan ada pengaruh jika Anda menonaktifkan optimisasi EBS.

Note

Anda juga dapat melihat informasi ini secara terprogram menggunakan AWS CLI Untuk informasi selengkapnya, lihat [Lihat tipe instans yang mendukung optimisasi EBS](#).

Topik

- [Tujuan umum](#)
- [Komputasi yang dioptimalkan](#)
- [Memori yang dioptimalkan](#)
- [Penyimpanan yang dioptimalkan](#)
- [Komputasi yang dipercepat](#)
- [Komputasi performa tinggi](#)

Tujuan umum

Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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	20000
a1.large ¹	525	3500	65.62	437.50	4000	20000
a1.xlarge ¹	800	3500	100.00	437.50	6000	20000
a1.2xlarge ¹	1750	3500	218.75	437.50	10000	20000
a1.4xlarge ²		3500		437.5		20000
a1.metal ²		3500		437.5		20000
m4.large ²		450		56,25		3600
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m5a.8xlarge ²		4750		593,75		20000
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
m5ad.4xlarge ²		2880		360.0		16000
m5ad.8xlarge ²		4750		593,75		20000
m5ad.12xlarge ²		6780		847.5		30000
m5ad.16xlarge ²		9500		1187,5		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
m5dn.metad 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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
m5zn.xlarge ¹	1564	3170	195,50	396.25	6667	13333
m5zn.2xlarge ²		3170		396.25		13333
m5zn.3xlarge ²		4750		593,75		20000
m5zn.6xlarge ²		9500		1187,5		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m5zn.12xlarge ²		19000		2375.0		80000
m5zn.medium 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	20000	40000
m6a.8xlarge ²		10000		1250.0		40000
m6a.12xlarge ²		15000		1875.0		60000
m6a.16xlarge ²		20000		2500.0		80000
m6a.24xlarge ²		30000		3750.0		120000
m6a.32xlarge ²		40000		5000.0		160000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6a.48xlarge ²		40000		5000.0		240000
m6a.metal ²		40000		5000.0		240000
m6g.medium ¹	315	4750	39,38	593,75	2500	20000
m6g.large ¹	630	4750	78.75	593,75	3600	20000
m6g.xlarge ¹	1188	4750	148,50	593,75	6000	20000
m6g.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
m6g.4xlarge ²		4750		593,75		20000
m6g.8xlarge ²		9500		1187,5		40000
m6g.12xlarge ²		14250		1781.25		50000
m6g.16xlarge ²		19000		2375.0		80000
m6g.metal ²		19000		2375.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6gd.medium ¹	315	4750	39,38	593,75	2500	20000
m6gd.large ¹	630	4750	78,75	593,75	3600	20000
m6gd.xlarge ¹	1188	4750	148,50	593,75	6000	20000
m6gd.2xlarge ¹	2375	4750	296,88	593,75	12000	20000
m6gd.4xlarge ²		4750		593,75		20000
m6gd.8xlarge ²		9500		1187,5		40000
m6gd.12xlarge ²		14250		1781,25		50000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6i.2xlarge ¹	2500	10000	312.50	1250,00	12000	40000
m6i.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
m6i.8xlarge ²		10000		1250.0		40000
m6i.12xlarge ²		15000		1875.0		60000
m6i.16xlarge ²		20000		2500.0		80000
m6i.24xlarge ²		30000		3750.0		120000
m6i.32xlarge ²		40000		5000.0		160000
m6i.metal ²		40000		5000.0		160000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6id.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
m6id.8xlarge ²		10000		1250.0		40000
m6id.12xlarge ²		15000		1875.0		60000
m6id.16xlarge ²		20000		2500.0		80000
m6id.24xlarge ²		30000		3750.0		120000
m6id.32xlarge ²		40000		5000.0		160000
m6id.metall ²		40000		5000.0		160000
m6idn.large ¹	1562	25000	195.31	3125.00	6250	100000
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	50000	100000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6idn.8xlarge ²		25000		3125.0		100000
m6idn.12xlarge ²		37500		4687.5		150000
m6idn.16xlarge ²		50000		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
m6in.2xlarge ¹	6250	25000	781,25	3125.00	25000	100000
m6in.4xlarge ¹	12500	25000	1562.50	3125.00	50000	100000
m6in.8xlarge ²		25000		3125.0		100000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m6in.12xlarge ²		37500		4687.5		150000
m6in.16xlarge ²		50000		6250.0		200000
m6in.24xlarge ²		75000		9375.0		300000
m6in.32xlarge ²		100000		12500.0		400000
m6in.meta1 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
m7a.2xlarge ¹	2500	10000	312.50	1250,00	12000	40000
m7a.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
m7a.8xlarge ²		10000		1250.0		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m7a.12xlarge ²		15000		1875.0		60000
m7a.16xlarge ²		20000		2500.0		80000
m7a.24xlarge ²		30000		3750.0		120000
m7a.32xlarge ²		40000		5000.0		160000
m7a.48xlarge ²		40000		5000.0		240000
m7a.metal-48xl ²		40000		5000.0		240000
m7g.medium ¹	315	10000	39,38	1250,00	2500	40000
m7g.large ¹	630	10000	78.75	1250,00	3600	40000
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	20000	40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m7g.8xlarge ²		10000		1250.0		40000
m7g.12xlarge ²		15000		1875.0		60000
m7g.16xlarge ²		20000		2500.0		80000
m7g.metal ²		20000		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
m7gd.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
m7gd.8xlarge ²		10000		1250.0		40000
m7gd.12xlarge ²		15000		1875.0		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m7gd.16xlarge ²		20000		2500.0		80000
m7gd.logam2		20000		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	20000	40000
m7i.8xlarge ²		10000		1250.0		40000
m7i.12xlarge ²		15000		1875.0		60000
m7i.16xlarge ²		20000		2500.0		80000
m7i.24xlarge ²		30000		3750.0		120000
m7i.48xlarge ²		40000		5000.0		240000
m7i.metal-24xl ²		30000		3750.0		120000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
m7i.metal-48xl ²		40000		5000.0		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	20000	40000
mac1.metal ²		14000		1750.0		80000
mac2.metal ²		10000		1250.0		55000
mac2-m2.metal ²		8000		1000,0		55000
mac2-m2pro.metal ²		8000		1000,0		55000
t3.nano ¹	43	2085	5.38	260.62	250	11800
t3.micro ¹	87	2085	10.88	260.62	500	11800

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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

Komputasi yang dioptimalkan

Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c4.4xlarge ²	2000		250.0		16000	
c4.8xlarge ²	4000		500,0		32000	
c5.large ¹	650	4750	81,25	593,75	4000	20000
c5.xlarge ¹	1150	4750	143,75	593,75	6000	20000
c5.2xlarge ¹	2300	4750	287.50	593,75	10000	20000
c5.4xlarge ²	4750		593,75		20000	
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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		20000
c5a.16xlarge ²		6300		787.5		26700
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c5ad.12xlarge ²	4750		593,75		20000	
c5ad.16xlarge ²	6300		787.5		26700	
c5ad.24xlarge ²	9500		1187,5		40000	
c5d.large ¹	650	4750	81,25	593,75	4000	20000
c5d.xlarge ¹	1150	4750	143,75	593,75	6000	20000
c5d.2xlarge ¹	2300	4750	287.50	593,75	10000	20000
c5d.4xlarge ²	4750		593,75		20000	
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	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c5n.large ¹	650	4750	81,25	593,75	4000	20000
c5n.xlarge ₁	1150	4750	143,75	593,75	6000	20000
c5n.2xlarge ¹	2300	4750	287.50	593,75	10000	20000
c5n.4xlarge ²		4750		593,75		20000
c5n.9xlarge ²		9500		1187,5		40000
c5n.18xlarge ²		19000		2375.0		80000
c5n.metal ²		19000		2375.0		80000
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	20000	40000
c6a.8xlarge ²		10000		1250.0		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6a.12xlarge ²		15000		1875.0		60000
c6a.16xlarge ²		20000		2500.0		80000
c6a.24xlarge ²		30000		3750.0		120000
c6a.32xlarge ²		40000		5000.0		160000
c6a.48xlarge ²		40000		5000.0		240000
c6a.metal ²		40000		5000.0		240000
c6g.medium ¹	315	4750	39,38	593,75	2500	20000
c6g.large ¹	630	4750	78.75	593,75	3600	20000
c6g.xlarge ¹	1188	4750	148,50	593,75	6000	20000
c6g.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
c6g.4xlarge ²		4750		593,75		20000
c6g.8xlarge ²		9500		1187,5		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6g.12xlarge ²		14250		1781.25		50000
c6g.16xlarge ²		19000		2375.0		80000
c6g.metal ²		19000		2375.0		80000
c6gd.medium ¹	315	4750	39,38	593,75	2500	20000
c6gd.large ¹	630	4750	78.75	593,75	3600	20000
c6gd.xlarge ¹	1188	4750	148,50	593,75	6000	20000
c6gd.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
c6gd.4xlarge ²		4750		593,75		20000
c6gd.8xlarge ²		9500		1187,5		40000
c6gd.12xlarge ²		14250		1781.25		50000
c6gd.16xlarge ²		19000		2375.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6gd.meta ¹		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	20000	40000
c6gn.4xlarge ²		9500		1187,5		40000
c6gn.8xlarge ²		19000		2375.0		80000
c6gn.12xlarge ²		28500		3562.5		120000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6i.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
c6i.8xlarge ²	10000		1250.0		40000	
c6i.12xlarge ²	15000		1875.0		60000	
c6i.16xlarge ²	20000		2500.0		80000	
c6i.24xlarge ²	30000		3750.0		120000	
c6i.32xlarge ²	40000		5000.0		160000	
c6i.metal ²	40000		5000.0		160000	
c6id.large ¹	650	10000	81,25	1250,00	3600	40000
c6id.xlarge ¹	1250	10000	156,25	1250,00	6000	40000
c6id.2xlarge ¹	2500	10000	312.50	1250,00	12000	40000
c6id.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
c6id.8xlarge ²	10000		1250.0		40000	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6id.12xlarge ²		15000		1875.0		60000
c6id.16xlarge ²		20000		2500.0		80000
c6id.24xlarge ²		30000		3750.0		120000
c6id.32xlarge ²		40000		5000.0		160000
c6id.metal ²		40000		5000.0		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	50000	100000
c6in.8xlarge ²		25000		3125.0		100000
c6in.12xlarge ²		37500		4687.5		150000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c6in.16xlarge ²		50000		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	20000	40000
c7a.8xlarge ²		10000		1250.0		40000
c7a.12xlarge ²		15000		1875.0		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c7a.16xlarge ²		20000		2500.0		80000
c7a.24xlarge ²		30000		3750.0		120000
c7a.32xlarge ²		40000		5000.0		160000
c7a.48xlarge ²		40000		5000.0		240000
c7a.metal-48xl ²		40000		5000.0		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	20000	40000
c7g.8xlarge ²		10000		1250.0		40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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		20000		2500.0		80000
c7g.metal ²		20000		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	20000	40000
c7gd.8xlarge ²		10000		1250.0		40000
c7gd.12xlarge ²		15000		1875.0		60000
c7gd.16xlarge ²		20000		2500.0		80000
c7gd.metal 2		20000		2500.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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	20000	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/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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	20000	40000
c7i.8xlarge ²		10000		1250.0		40000
c7i.12xlarge ²		15000		1875.0		60000
c7i.16xlarge ²		20000		2500.0		80000
c7i.24xlarge ²		30000		3750.0		120000
c7i.48xlarge ²		40000		5000.0		240000
c7i.metal-24xl ²		30000		3750.0		120000
c7i.metal-48xl ²		40000		5000.0		240000
c7i-flex.large ¹	312	10000	39,06	1250,00	2500	40000
c7i-flex.xlarge ¹	625	10000	78.12	1250,00	3600	40000
c7i-flex.2xlarge ¹	1250	10000	156,25	1250,00	6000	40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
c7i-flex.4xlarge 1	2500	10000	312.50	1250,00	12000	40000
c7i-flex.8xlarge 1	5000	10000	625.00	1250,00	20000	40000

Memori yang dioptimalkan

Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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		20000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r5ad.8xlarge ²	4750		593,75		20000	
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
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	20000		2500.0		86667	
r5b.12xlarge ²	30000		3750.0		130000	
r5b.16xlarge ²	40000		5000.0		173333	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
r5dn.24xlarge ²		19000		2375.0		80000
r5dn.meta1 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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
r6a.2xlarge ¹	2500	10000	312.50	1250,00	12000	40000
r6a.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
r6a.8xlarge ²		10000		1250.0		40000
r6a.12xlarge ²		15000		1875.0		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r6a.16xlarge ²		20000		2500.0		80000
r6a.24xlarge ²		30000		3750.0		120000
r6a.32xlarge ²		40000		5000.0		160000
r6a.48xlarge ²		40000		5000.0		240000
r6a.metal ²		40000		5000.0		240000
r6g.medium ¹	315	4750	39,38	593,75	2500	20000
r6g.large ¹	630	4750	78.75	593,75	3600	20000
r6g.xlarge ¹	1188	4750	148,50	593,75	6000	20000
r6g.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
r6g.4xlarge ²		4750		593,75		20000
r6g.8xlarge ²		9500		1187,5		40000
r6g.12xlarge ²		14250		1781.25		50000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r6g.16xlarge ²		19000		2375.0		80000
r6g.metal ²		19000		2375.0		80000
r6gd.medium ¹	315	4750	39,38	593,75	2500	20000
r6gd.large ¹	630	4750	78.75	593,75	3600	20000
r6gd.xlarge ¹	1188	4750	148,50	593,75	6000	20000
r6gd.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
r6gd.4xlarge ²		4750		593,75		20000
r6gd.8xlarge ²		9500		1187,5		40000
r6gd.12xlarge ²		14250		1781.25		50000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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	20000	40000
r6i.8xlarge ²		10000		1250.0		40000
r6i.12xlarge ²		15000		1875.0		60000
r6i.16xlarge ²		20000		2500.0		80000
r6i.24xlarge ²		30000		3750.0		120000
r6i.32xlarge ²		40000		5000.0		160000
r6i.metal ²		40000		5000.0		160000
r6idn.large ¹	1562	25000	195.31	3125.00	6250	100000
r6idn.xlarge ¹	3125	25000	390.62	3125.00	12500	100000
r6idn.2xlarge ¹	6250	25000	781,25	3125.00	25000	100000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r6idn.4xlarge ¹	12500	25000	1562.50	3125.00	50000	100000
r6idn.8xlarge ²	25000	50000	3125.0	6250.0	100000	200000
r6idn.12xlarge ²	37500	75000	4687.5	9375.0	150000	300000
r6idn.16xlarge ²	50000	100000	6250.0	12500.0	200000	400000
r6idn.24xlarge ²	75000	150000	9375.0	18750.0	300000	600000
r6idn.32xlarge ²	100000	200000	12500.0	25000.0	400000	800000
r6idn.metal ²	100000	200000	12500.0	25000.0	400000	800000
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	50000	100000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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		50000		6250.0		200000
r6in.24xlarge ²		75000		9375.0		300000
r6in.32xlarge ²		100000		12500.0		400000
r6in.metal ²		100000		12500.0		400000
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	20000	40000
r6id.8xlarge ²		10000		1250.0		40000
r6id.12xlarge ²		15000		1875.0		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r6id.16xlarge ²		20000		2500.0		80000
r6id.24xlarge ²		30000		3750.0		120000
r6id.32xlarge ²		40000		5000.0		160000
r6id.metal ²		40000		5000.0		160000
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	20000	40000
r7a.8xlarge ₂		10000		1250.0		40000
r7a.12xlarge ²		15000		1875.0		60000
r7a.16xlarge ²		20000		2500.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r7a.24xlarge ²		30000		3750.0		120000
r7a.32xlarge ²		40000		5000.0		160000
r7a.48xlarge ²		40000		5000.0		240000
r7a.metal-48xl ²		40000		5000.0		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	20000	40000
r7g.8xlarge ₂		10000		1250.0		40000
r7g.12xlarge ²		15000		1875.0		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r7g.16xlarge ²		20000		2500.0		80000
r7g.metal ²		20000		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	20000	40000
r7gd.8xlarge ²		10000		1250.0		40000
r7gd.12xlarge ²		15000		1875.0		60000
r7gd.16xlarge ²		20000		2500.0		80000
r7gd.logam ²		20000		2500.0		80000
r7i.large ¹	650	10000	81,25	1250,00	3600	40000
r7i.xlarge ¹	1250	10000	156,25	1250,00	6000	40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
r7i.2xlarge ¹	2500	10000	312.50	1250,00	12000	40000
r7i.4xlarge ¹	5000	10000	625.00	1250,00	20000	40000
r7i.8xlarge ²		10000		1250.0		40000
r7i.12xlarge ²		15000		1875.0		60000
r7i.16xlarge ²		20000		2500.0		80000
r7i.24xlarge ²		30000		3750.0		120000
r7i.48xlarge ²		40000		5000.0		240000
r7i.metal-24xl ²		30000		3750.0		120000
r7i.metal-48xl ²		40000		5000.0		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/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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	20000	40000
r7iz.8xlarge ²		10000		1250.0		40000
r7iz.12xlarge ²		19000		2375.0		76000
r7iz.16xlarge ²		20000		2500.0		80000
r7iz.32xlarge ²		40000		5000.0		160000
r7iz.meta-l-16xl ²		20000		2500.0		80000
r7iz.meta-l-32xl ²		40000		5000.0		160000
u-3tb1.56xlarge ²		19000		2375.0		80000
u-6tb1.56xlarge ²		38000		4750,0		160000
u-6tb1.112xlarge ²		38000		4750,0		160000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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-12tb.224xbesar ²	60000		7500,0		420000	
u7in-16tb.224xlarge ²	100000		12500,0		420000	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
u7in-24tb .224xlarge 2		100000		12500.0		420000
u7in-32tb .224xlarge 2		100000		12500.0		420000
x1.16xlarge ²		7000		875.0		40000
x1.32xlarge ²		14000		1750.0		80000
x2gd.medium ¹	315	4750	39,38	593,75	2500	20000
x2gd.large ¹	630	4750	78.75	593,75	3600	20000
x2gd.xlarge ¹	1188	4750	148,50	593,75	6000	20000
x2gd.2xlarge ¹	2375	4750	296.88	593,75	12000	20000
x2gd.4xlarge ²		4750		593,75		20000
x2gd.8xlarge ²		9500		1187,5		40000
x2gd.12xlarge ²		14250		1781.25		60000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
x2gd.16xlarge ²		19000		2375.0		80000
x2gd.metal ²		19000		2375.0		80000
x2idn.16xlarge ²		40000		5000.0		173333
x2idn.24xlarge ²		60000		7500.0		260000
x2idn.32xlarge ²		80000		10000.0		260000
x2idn.metal ²		80000		10000.0		260000
x2iedn.xlarge ¹	2500	20000	312.50	2500.00	8125	65000
x2iedn.2xlarge ¹	5000	20000	625.00	2500.00	16250	65000
x2iedn.4xlarge ¹	10000	20000	1250,00	2500.00	32500	65000
x2iedn.8xlarge ²		20000		2500.0		65000
x2iedn.16xlarge ²		40000		5000.0		130000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
x2iedn.24xlarge ²	60000		7500.0		195000	
x2iedn.32xlarge ²	80000		10000.0		260000	
x2iedn.metal ²	80000		10000.0		260000	
x2iezn.2xlarge ²	3170		396.25		13333	
x2iezn.4xlarge ²	4750		593,75		20000	
x2iezn.6xlarge ²	9500		1187,5		40000	
x2iezn.8xlarge ²	12000		1500.0		55000	
x2iezn.12xlarge ²	19000		2375.0		80000	
x2iezn.metal ²	19000		2375.0		80000	
x1e.xlarge ²	500		62.5		3700	
x1e.2xlarge ²	1000		125,0		7400	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
x1e.4xlarge ²		1750		218.75		10000
x1e.8xlarge ²		3500		437.5		20000
x1e.16xlarge ²		7000		875.0		40000
x1e.32xlarge ²		14000		1750.0		80000
z1d.large ¹	800	3170	100.00	396.25	3333	13333
z1d.xlarge ₁	1580	3170	197.50	396.25	6667	13333
z1d.2xlarge ²		3170		396.25		13333
z1d.3xlarge ²		4750		593,75		20000
z1d.6xlarge ²		9500		1187,5		40000
z1d.12xlarge ²		19000		2375.0		80000
z1d.metal ²		19000		2375.0		80000

Penyimpanan yang dioptimalkan

⚠ Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
d2.xlarge ²		750		93,75		6000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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
d3en.12xlarge ²		7000		875.0		40000
h1.2xlarge ²		1750		218.75		12000
h1.4xlarge ²		3500		437.5		20000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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	20000
i3en.xlarge ¹	1153	4750	144,20	593,75	6000	20000
i3en.2xlarge ¹	2307	4750	288.39	593,75	12000	20000
i3en.3xlarge ¹	3800	4750	475.00	593,75	15000	20000
i3en.6xlarge ²		4750		593,75		20000
i3en.12xlarge ²		9500		1187,5		40000
i3en.24xlarge ²		19000		2375.0		80000
i3en.metal ²		19000		2375.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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	20000	40000
i4g.8xlarge ²		10000		1250.0		40000
i4g.16xlarge ²		20000		2500.0		80000
i4i.large ¹	625	10000	78.12	1250,00	2500	40000
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	20000	40000
i4i.8xlarge ²		10000		1250.0		40000
i4i.12xlarge ²		15000		1875.0		60000
i4i.16xlarge ²		20000		2500.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
i4i.24xlarge ²		30000		3750.0		120000
i4i.32xlarge ²		40000		5000.0		160000
i4i.metal ²		40000		5000.0		160000
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	20000	40000
im4gn.4xlarge ²		10000		1250.0		40000
im4gn.8xlarge ²		20000		2500.0		80000
im4gn.16xlarge ²		40000		5000.0		160000
is4gen.medium ¹	625	10000	78.12	1250,00	2500	40000
is4gen.large ¹	1250	10000	156,25	1250,00	5000	40000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
is4gen.xlarge ¹	2500	10000	312.50	1250,00	10000	40000
is4gen.2xlarge ¹	5000	10000	625.00	1250,00	20000	40000
is4gen.4xlarge ²		10000		1250.0		40000
is4gen.8xlarge ²		20000		2500.0		80000

Komputasi yang dipercepat

Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
dl1.24xlarge ²		19000		2375.0		80000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
dl2q.24xlarge ²		19000		2375.0		80000
f1.2xlarge ²		1700		212.5		12000
f1.4xlarge ²		3500		437.5		44000
f1.16xlarge ²		14000		1750.0		75000
g3.4xlarge ²		3500		437.5		20000
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
g4ad.8xlarge ²		3170		396.25		13333
g4ad.16xlarge ²		6300		787,5		26667

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
g4dn.xlarge ¹	950	3500	118.75	437.50	3000	20000
g4dn.2xlarge ¹	1150	3500	143,75	437.50	6000	20000
g4dn.4xlarge ²		4750		593,75		20000
g4dn.8xlarge ²		9500		1187,5		40000
g4dn.12xlarge ²		9500		1187,5		40000
g4dn.16xlarge ²		9500		1187,5		40000
g4dn.meta1 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		20000
g5.8xlarge ²		16000		2000,0		65000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
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	20000
g5g.2xlarge _{ge¹}	2375	4750	296.88	593,75	12000	20000
g5g.4xlarge _{ge²}		4750		593,75		20000
g5g.8xlarge _{ge²}		9500		1187,5		40000
g5g.16xlarge _{rge²}		19000		2375.0		80000
g5g.metal ²		19000		2375.0		80000
g6.xlarge ₁	1000	5000	125.00	625.00	4000	20000
g6.2xlarge ₁	2000	5000	250.00	625.00	8000	20000
g6.4xlarge ₂		8000		1000,0		32000

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
g6.8xlarge 2		16000		2000,0		64000
g6.12xlarge 2		20000		2500.0		80000
g6.16xlarge 2		20000		2500.0		80000
g6.24xlarge 2		30000		3750.0		120000
g6.48xlarge 2		60000		7500.0		240000
gr6.4xbesar 2		8000		1000,0		32000
gr6.8xbesar 2		16000		2000,0		64000
inf1.xlarge 1	1190	4750	148,75	593,75	4000	20000
inf1.2xlarge 1	1190	4750	148,75	593,75	6000	20000
inf1.6xlarge 2		4750		593,75		20000
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

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
inf2.48xlarge ²	60000		7500.0		240000	
p2.xlarge ²	750		93,75		6000	
p2.8xlarge ₂	5000		625.0		32500	
p2.16xlarge ²	10000		1250.0		65000	
p3.2xlarge ₂	1750		218.75		10000	
p3.8xlarge ₂	7000		875.0		40000	
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	
p5.48xlarge ²	80000		10000.0		260000	

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS dasar (16 KiB I/O)	IOPS maksimum (16 KiB I/O)
trn1.2xlarge ¹	5000	20000	625.00	2500.00	16250	65000
trn1.32xlarge ²		80000		10000.0		260000
trn1n.32xlarge ²		80000		10000.0		260000
vt1.3xlarge ¹	2375	4750	296.88	593,75	10000	20000
vt1.6xlarge ²		4750		593,75		20000
vt1.24xlarge ²		19000		2375.0		80000

Komputasi performa tinggi

Important

¹ Instans ini dapat mendukung performa maksimum selama 30 menit setidaknya setiap 24 jam sekali, setelah itu, instans kembali ke performa garis acuan.

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.

Ukuran instans	Bandwidth dasar (Mbps)	Bandwidth maksimum (Mbps)	Throughput dasar (MB/dtk, 128 KiB I/O)	Throughput maksimum (MB/dtk, 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

Optimisasi EBS didukung

Tabel berikut mencantumkan tipe instans yang mendukung optimisasi EBS, tetapi optimisasi EBS tidak diaktifkan secara default. Anda dapat mengaktifkan optimisasi EBS saat Anda meluncurkan instans ini atau setelah proses berjalan. Optimisasi EBS instans harus diaktifkan untuk mencapai

tingkat performa yang dijelaskan. Saat Anda mengaktifkan optimisasi EBS untuk instans yang tidak dioptimalkan EBS secara default, Anda membayar biaya per jam tambahan yang rendah untuk kapasitas khusus. Untuk informasi harga, lihat [Instans yang Dioptimalkan EBS di Halaman Harga Sesuai Permintaan, Harga Amazon EC2](#).

Note

Anda juga dapat melihat informasi ini secara terprogram menggunakan AWS CLI Untuk informasi selengkapnya, lihat [Lihat tipe instans yang mendukung optimisasi EBS](#).

Ukuran instans	Bandwidth maksimum (Mbps)	Throughput maksimum (MB/dtk, 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

Ukuran instans	Bandwidth maksimum (Mbps)	Throughput maksimum (MB/dtk, 128 KiB I/O)	IOPS maksimum (16 KiB I/O)
r3.xlarge	500	62.5	4000
r3.2xlarge	1000	125,0	8000
r3.4xlarge	2000	250.0	16000

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 performa maksimum

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 menyetel 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 [Pantau 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. Begitu juga, untuk beban kerja dengan throughput tinggi, gunakan enam volume `io1` atau `io2` dengan 48.000 IOPS yang Tersedia untuk mendapatkan throughput maksimum 4.750 MB/dtk. Untuk rekomendasi tambahan, lihat [Konfigurasi Penyimpanan untuk SAP HANA](#).

Pertimbangan

- Instans `G4dn`, `I3en`, `Inf1`, `M5a`, `M5ad`, `R5a`, `R5ad`, `T3`, `T3a`, dan `Z1d` yang diluncurkan setelah 26 Februari 2020 memberikan performa maksimal yang tercantum dalam tabel di atas. 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 performa maksimum yang tercantum dalam tabel di atas. Untuk mendapatkan performa maksimum dari instans yang diluncurkan sebelum 3 Desember 2019, hentikan dan mulai.
- Instans u-6tb1.metal, u-9tb1.metal, dan u-12tb1.metal yang diluncurkan setelah 12 Maret 2020 memberikan performa dalam tabel di atas. 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 memperbarui instansnya tanpa biaya tambahan.

Lihat tipe instans yang mendukung optimisasi EBS

Anda dapat menggunakan AWS CLI untuk melihat jenis instans di Wilayah saat ini yang mendukung pengoptimalan EBS.

Untuk melihat tipe instans yang mendukung optimisasi EBS dan yang telah diaktifkan secara default

Gunakan perintah perintah [describe-instance-types](#) berikut ini. Jika menjalankan perintah ini di Windows Command Prompt, ganti karakter kelanjutan baris \ dengan karakter ^.

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIOPS,"MaxThroughput(MB/s)":EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=default --output=table
```

Output contoh untuk eu-west-1:

```
-----
|                               DescribeInstanceTypes                               |
+-----+-----+-----+-----+
| InstanceType | MaxBandwidth(Mb/s) | MaxIOPS | MaxThroughput(MB/s) |
+-----+-----+-----+-----+
| m5dn.8xlarge | 6800                | 30000  | 850.0               |
| m6gd.xlarge  | 4750                | 20000  | 593.75              |
| c4.4xlarge   | 2000                | 16000  | 250.0               |
| r4.16xlarge  | 14000              | 75000  | 1750.0              |
-----
```

```
| m5ad.large | 2880 | 16000 | 360.0 |
...

```

Untuk melihat tipe instans yang mendukung optimisasi EBS dan yang telah diaktifkan secara default

Gunakan perintah `describe-instance-types` berikut ini.

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIOPS):EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=supported --output=table

```

Output contoh 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 |
+-----+-----+-----+-----+

```

Aktifkan optimisasi EBS saat peluncuran

Anda dapat mengaktifkan optimisasi untuk sebuah instans dengan mengatur atribut untuk optimisasi EBS.

Untuk memungkinkan optimisasi Amazon EBS saat meluncurkan instans menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pilih Luncurkan Instans.
3. Dalam Langkah 1: Pilih Amazon Machine Image (AMI), pilih AMI.
4. Dalam Langkah 2: Pilih Tipe Instans, pilih tipe instans yang tercantum sebagai mendukung optimisasi Amazon EBS.
5. Dalam Langkah 3: Konfigurasi Rincian Instans, lengkapi bidang yang Anda butuhkan dan pilih Luncurkan sebagai instans yang dioptimalkan untuk EBS. Jika tipe instans yang Anda pilih pada langkah sebelumnya tidak mendukung optimisasi Amazon EBS, opsi ini tidak tersedia. Apabila tipe instans yang Anda pilih adalah Amazon EBS secara default, opsi ini dipilih dan Anda tidak dapat membatalkannya.
6. Ikuti petunjuk untuk menyelesaikan wizard dan meluncurkan instans Anda.

Untuk mengaktifkan optimisasi EBS saat meluncurkan instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut dengan opsi yang sesuai. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [run-instances](#) dengan `--ebs-optimized` (AWS CLI)
- [New-EC2Instance](#) dengan `-EbsOptimized` (AWS Tools for Windows PowerShell)

Aktifkan optimisasi EBS untuk instans yang sudah ada

Anda dapat mengaktifkan atau menonaktifkan optimisasi untuk instans yang sudah ada dengan mengubah atribut instans yang dioptimalkan Amazon EBS. Jika instans sedang berjalan, Anda harus menghentikannya terlebih dahulu.

Warning

Ketika Anda menghentikan suatu instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan instans, pastikan untuk mencadangkannya ke penyimpanan persisten.

Untuk mengaktifkan optimisasi EBS untuk instans yang sudah ada menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans, dan pilih instans.
3. Untuk menghentikan instans, pilih Tindakan, Status instans, Hentikan instans. Tindakan ini dapat memakan waktu beberapa menit sampai instans berhenti.
4. Dengan instans yang masih dipilih, klik Tindakan, Pengaturan instans, Ubah tipe instans.
5. Untuk Ubah tipe instans, lakukan salah satu hal berikut:
 - Jika tipe instans Anda adalah yang dioptimalkan Amazon EBS secara default, Dioptimalkan EBS dipilih dan Anda tidak dapat mengubahnya. Anda dapat memilih Batalkan, karena optimisasi Amazon EBS sudah diaktifkan untuk instans tersebut.
 - Jika tipe instans Anda mendukung optimisasi Amazon EBS, pilih Dioptimalkan EBS lalu pilih Terapkan.
 - Jika tipe instans Anda tidak mendukung optimisasi Amazon EBS, Anda tidak dapat memilih Dioptimalkan EBS. Anda dapat memilih tipe instans dari tipe instans yang mendukung optimisasi Amazon EBS, pilih Dioptimalkan EBS, lalu pilih Terapkan.
6. Pilih Status instans, Mulai instans.

Untuk mengaktifkan optimisasi EBS untuk instans yang sudah ada menggunakan baris perintah

1. Jika instans sedang berjalan, gunakan salah satu perintah berikut untuk menghentikannya:
 - [stop-instances](#) (AWS CLI)
 - [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell)
2. Untuk mengaktifkan optimisasi EBS, gunakan salah satu perintah berikut dengan opsi terkait:
 - [modify-instance-attribute](#) dengan `--ebs-optimized` (AWS CLI)
 - [Edit-EC2InstanceAttribute](#) dengan `-EbsOptimized` (AWS Tools for Windows PowerShell)

Opsi pembelian instans

Amazon EC2 menyediakan opsi pembelian berikut agar Anda dapat mengoptimalkan biaya berdasarkan kebutuhan Anda:

- [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 kluster 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 [Penetapan Harga Amazon EC2](#).

Untuk mempelajari Savings Plans selengkapnya, lihat [Panduan Pengguna Savings Plans](#).

Daftar Isi

- [Menentukan siklus hidup instans](#)
- [Instans Sesuai Permintaan](#)
- [Instans Terpesan](#)
- [Instans Spot](#)
- [Host Khusus](#)
- [Instans Khusus](#)
- [Reservasi Kapasitas](#)

Menentukan siklus hidup instans

Siklus hidup sebuah instans dimulai saat diluncurkan dan berakhir saat diakhiri. Opsi pembelian yang Anda pilih memengaruhi siklus hidup instans. Misalnya, Instans Sesuai Permintaan berjalan saat Anda meluncurkannya dan berakhir saat Anda mengakhirinya. Instans Spot berjalan selama kapasitas tersedia dan harga maksimum Anda lebih tinggi dari harga Spot.

Gunakan salah satu metode berikut untuk menentukan siklus hidup sebuah instans.

Untuk menentukan siklus hidup instans menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Di tab Detail, pada Detail instans, temukan Siklus Hidup. Jika nilainya spot, instans tersebut adalah Instans Spot. Jika nilainya normal, instans tersebut bisa berupa Instans Sesuai Permintaan atau Instans Terpesan.
5. Di tab Detail, pada Host dan grup penempatan, temukan Penghunian. Jika nilainya host, instans berjalan pada Host Khusus. Jika nilainya dedicated, instans tersebut adalah Instans Khusus.

Untuk menentukan siklus hidup instance menggunakan AWS CLI

Gunakan perintah [describe-instances](#) berikut ini:

```
aws ec2 describe-instances --instance-ids i-1234567890abcdef0
```

Jika instans berjalan di Host Khusus, output berisi informasi berikut ini:

```
"Tenancy": "host"
```

Jika instans adalah Instans Khusus, output berisi informasi berikut ini:

```
"Tenancy": "dedicated"
```

Jika instans adalah Instans Spot, output berisi informasi berikut ini:

```
"InstanceLifecycle": "spot"
```

Atau, output tidak berisi InstanceLifecycle.

Instans Sesuai Permintaan

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 sudah tetap, dan tercantum di [halaman Harga Amazon EC2, halaman Harga Sesuai Permintaan](#).

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 [Instans Terpesan](#).

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 dalam hal jumlah unit pemrosesan pusat virtual (vCPU) yang digunakan Instans Sesuai Permintaan Anda, apa pun tipe instansnya. Setiap jenis kuota menentukan jumlah maksimum vCPU untuk satu atau beberapa keluarga instans.

Akun Anda menyertakan kuota berikut untuk Instans Sesuai Permintaan. Kuota hanya berlaku untuk menjalankan instans. Jika instans Anda tertunda, berhenti, berhenti, atau hibernasi, itu tidak dihitung dalam kuota Anda.

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 Instans Amazon EC2](#).

Anda dapat meluncurkan kombinasi tipe instans apa pun yang memenuhi kebutuhan aplikasi Anda yang terus berubah, selama jumlah vCPU tidak melebihi kuota akun Anda. Sebagai contoh, dengan kuota instans Standar 256 vCPU, Anda dapat meluncurkan 32 instans `m5.2xlarge` (32 x 8 vCPU) atau 16 instans `c5.4xlarge` (16 x 16 vCPU). Untuk informasi selengkapnya, lihat [Batas Instans Sesuai Permintaan EC2](#).

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 Kuota Layanan 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 vCPU untuk setiap tipe 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 penggunaan EC2 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, 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.

Instans Terpesan

Important

Kami merekomendasikan Savings Plans atas Instans Cadangan. Paket Tabungan adalah cara termudah dan paling fleksibel untuk menghemat uang pada biaya AWS komputasi Anda dan menawarkan harga yang lebih rendah (diskon hingga 72% dari harga On-Demand), seperti Instans Cadangan. Namun, Savings Plans berbeda dengan Instans Cadangan. Dengan Instans Cadangan, Anda berkomitmen pada konfigurasi instans tertentu, sedangkan dengan Savings Plans, Anda memiliki fleksibilitas untuk menggunakan konfigurasi instans yang paling sesuai dengan kebutuhan Anda. Untuk menggunakan Savings Plans, Anda membuat komitmen terhadap jumlah penggunaan yang konsisten, diukur dalam USD per jam. Untuk informasi selengkapnya, lihat [AWS Panduan Pengguna Savings Plans](#).

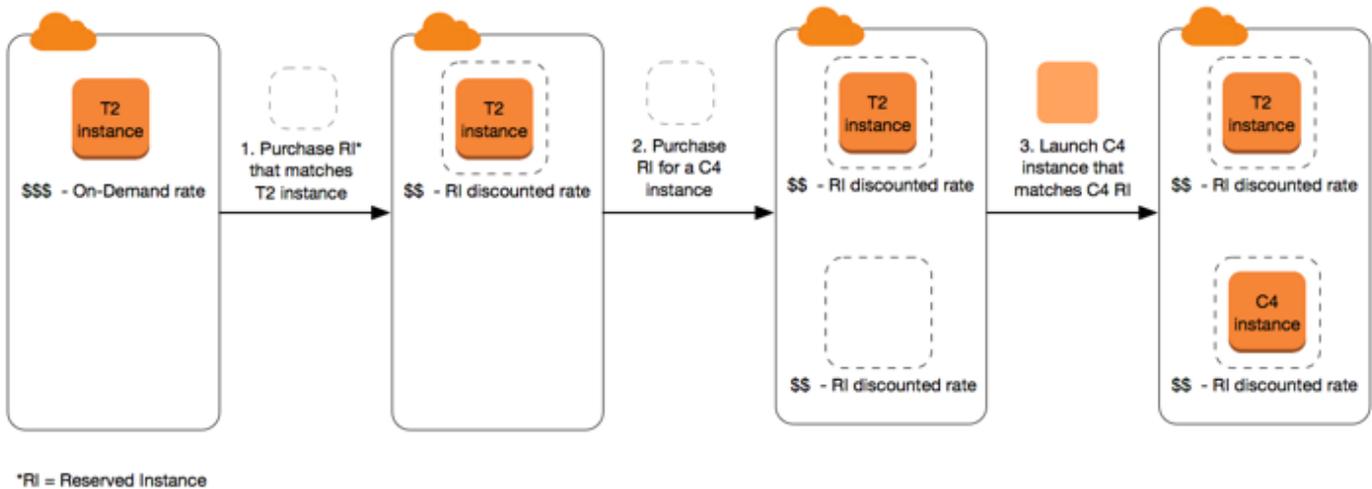
Instans Terpesan memberi Anda penghematan yang signifikan pada biaya Amazon EC2 Anda dibandingkan dengan harga Instans Sesuai Permintaan. Instans Terpesan bukanlah instans fisik, melainkan diskon penagihan yang diterapkan untuk penggunaan Instans Sesuai Permintaan di akun Anda. Instans Sesuai Permintaan ini harus cocok dengan atribut tertentu, seperti tipe instans dan Wilayah, untuk mendapatkan keuntungan dari diskon penagihan.

Topik Instans Terpesan

- [Ringkasan Instans Terpesan](#)
- [Variabel utama yang menentukan harga Instans Terpesan](#)
- [Instans Terpesan Regional dan zonal \(cakupan\)](#)
- [Tipe Instans Terpesan \(kelas penawaran\)](#)
- [Bagaimana Instans Terpesan diterapkan](#)
- [Menggunakan Instans Terpesan Anda](#)
- [Bagaimana Anda ditagih](#)
- [Membeli Instans Terpesan](#)
- [Menjual di Marketplace Instans Terpesan](#)
- [Memodifikasi Instans Terpesan](#)
- [Menukar Instans Terpesan Konvertibel](#)
- [Kuota Instans Terpesan](#)

Ringkasan Instans Terpesan

Diagram berikut menunjukkan gambaran umum dasar pembelian dan penggunaan Instans Terpesan.



Dalam skenario ini, Anda memiliki Instans Sesuai Permintaan (T2) yang berjalan di akun Anda, yang saat ini Anda bayar dengan tarif Sesuai Permintaan. 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](#).
- 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.

- Satu tahun: Satu tahun didefinisikan sebagai 31.536.000 detik (365 hari).
- Tiga tahun: Tiga tahun didefinisikan sebagai 94.608.000 detik (1.095 hari).

Instans Terpesan tidak diperpanjang secara otomatis; saat kedaluwarsa, Anda dapat terus menggunakan instans EC2 tanpa gangguan, tetapi Anda dikenai tarif Sesuai Permintaan. Dalam contoh di atas, ketika Instans Terpesan yang mencakup instans T2 dan C4 telah kedaluwarsa, Anda kembali membayar tarif Sesuai Permintaan hingga Anda mengakhiri instans atau membeli Instans Terpesan baru yang cocok dengan atribut instans.

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 [Menjual di Marketplace Instans Terpesan](#).

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.
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. Hanya didukung di Instans Terpesan Amazon Linux/Uni	Tidak ada fleksibilitas ukuran instans — diskon Instans Terpesan berlaku untuk penggunaan instans dengan tipe dan ukuran instans yang ditentukan.

	Instans Terpesan Regional	Instans Terpesan Zonal
	x dengan penghunian default. Untuk informasi selengkapnya, lihat Fleksibilitas ukuran instans ditentukan oleh faktor normalisasi .	
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 Instans Terpesan 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.

Perbedaan antara Instans Terpesan Standar dan Konvertibel

Berikut adalah perbedaan 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

	Instans Terpesan Standar	Instans Terpesan Konvertibel
		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 Instans Terpesan diterapkan

Instans Terpesan bukanlah instans fisik, melainkan diskon penagihan yang diterapkan untuk Instans Sesuai Permintaan yang berjalan di akun Anda. Instans Sesuai Permintaan harus cocok dengan spesifikasi Instans Terpesan tertentu untuk mendapatkan keuntungan dari diskon penagihan.

Jika Anda membeli Instans Terpesan dan sudah memiliki Instans Sesuai Permintaan yang berjalan yang sesuai dengan spesifikasi Instans Terpesan, diskon penagihan langsung diterapkan secara otomatis. Anda tidak perlu memulai ulang instans Anda. Jika Anda tidak memiliki Instans Sesuai Permintaan yang memenuhi syarat, luncurkan Instans Sesuai Permintaan dengan spesifikasi yang sama dengan Instans Terpesan 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 (penghunian, platform, Zona Ketersediaan, tipe instans, dan ukuran instans) dari instans yang sedang berjalan harus cocok dengan Instans Terpesan.

Misalnya, jika Anda membeli dua penghunian default `c4.xlarge` Instans Terpesan Standar Linux/Unix untuk Zona Ketersediaan `us-east-1a`, maka maksimal dua penghunian default `c4.xlarge` instans Linux/Unix yang berjalan di Zona Ketersediaan `us-east-1a` yang dapat memanfaatkan diskon Instans Terpesan.

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 Terpesan berlaku untuk penggunaan instans untuk instans yang memiliki [keluarga, generasi, dan atribut](#) 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, Inf1, dan Inf2
- 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

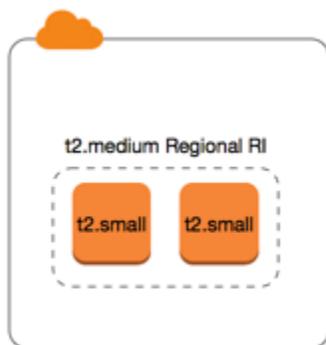
Fleksibilitas ukuran instans ditentukan oleh faktor normalisasi ukuran instans. Diskon berlaku baik sepenuhnya atau sebagian untuk instans yang berjalan dari keluarga instans yang sama, bergantung pada ukuran instans reservasi, di Zona Ketersediaan mana pun di Wilayah itu. Atribut yang harus dicocokkan 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.

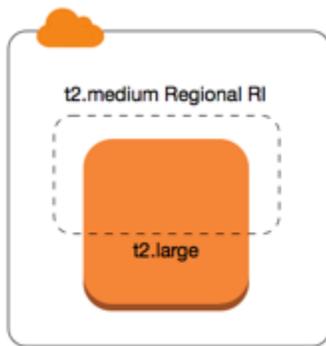
Ukuran instans	Faktor normalisasi
nano	0,25
micro	0,5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
3xlarge	24
4xlarge	32
6xlarge	48

Ukuran instans	Faktor normalisasi
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256
48xlarge	384
56xlarge	448
112xlarge	896

Misalnya, instans `t2.medium` memiliki faktor normalisasi 2. Jika Anda membeli Instans Terpesan Amazon Linux/Unix penghunian default `t2.medium` di AS Timur (Virginia Utara) dan Anda memiliki dua instans `t2.small` yang sedang berjalan di akun Anda di Wilayah itu, manfaat penagihan diterapkan secara penuh untuk 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 Terpesan regional Amazon Linux/Unix dengan penghunian bersama pada instans bare metal, Anda dapat memanfaatkan penghematan Instans Terpesan dalam keluarga instans yang sama. Berlaku juga sebaliknya: jika Anda memiliki Instans Terpesan regional Amazon Linux/Unix dengan penghunian bersama pada instans dalam keluarga yang sama dengan instans bare metal, Anda dapat memanfaatkan penghematan Instans Terpesan 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
<code>a1.metal</code>	32
<code>m5zn.metal</code> <code>x2iezn.metal</code> <code>z1d.metal</code>	96
<code>c6g.metal</code> <code>c6gd.metal</code> <code>i3.metal</code> <code>m6g.metal</code> <code>m6gd.metal</code> <code>r6g.metal</code> <code>r6gd.metal</code> <code>x2gd.metal</code>	128
<code>c5n.metal</code>	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-*.metal	896

Misalnya, `i3.metal` Misalnya memiliki faktor normalisasi 128. Jika Anda membeli Instans Terpesan Amazon Linux/Unix penghunian default `i3.metal` 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 Terpesan Amazon Linux/Unix penghunian default `i3.8xlarge` di AS Timur (Virginia Utara), Anda memiliki satu instans `i3.metal` yang berjalan di Wilayah tersebut, keuntungan penagihan diterapkan secara penuh ke instans `i3.metal`.

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 Instans Sesuai Permintaan 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.

Sebuah instans m4.large setara dengan 4 unit/jam yang dinormalisasi.

Anda telah membeli empat Instans Terpesan regional m4.large, dan totalnya sama dengan 16 unit/jam yang dinormalisasi (4x4). Akun A memiliki dua instans m4.xlarge yang berjalan, yang setara dengan 16 unit/jam yang dinormalisasi (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`. Satu instans `c4.large` setara dengan 4 unit/jam yang dinormalisasi dan satu `c4.xlarge` setara dengan 8 unit/jam yang dinormalisasi.

Dalam hal ini, Instans Terpesan regional `c4.large` memberikan sebagian keuntungan untuk penggunaan `c4.xlarge`. Ini karena Instans Terpesan `c4.large` setara dengan 4 unit/jam penggunaan yang dinormalisasi, tetapi instans `c4.xlarge` membutuhkan 8 unit/jam yang dinormalisasi. Oleh karena itu, diskon penagihan Instans Terpesan `c4.large` berlaku untuk 50% dari penggunaan `c4.xlarge`. Penggunaan `c4.xlarge` yang tersisa dikenai biaya dengan tarif Sesuai Permintaan.

Skenario 2: Instans Terpesan dalam satu akun menggunakan faktor normalisasi

Anda menjalankan Instans Sesuai Permintaan 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 instans `m3.large` setara dengan 4 unit/jam yang dinormalisasi.

Sebuah instans `m3.xlarge` setara dengan 8 unit/jam yang dinormalisasi.

Sebuah instans `m3.2xlarge` setara dengan 16 unit/jam yang dinormalisasi.

Manfaatnya diterapkan sebagai berikut:

Instans Terpesan regional `m3.2xlarge` memberikan keuntungan penuh untuk penggunaan 2 x `m3.large`, karena kedua instans ini mencakup 8 unit/jam yang dinormalisasi. Hal ini menyisakan 8 unit/jam yang dinormalisasi untuk diterapkan pada instans `m3.xlarge`.

Dengan sisa 8 unit/jam yang dinormalisasi, Instans Terpesan regional `m3.2xlarge` memberikan manfaat penuh untuk 1 x penggunaan `m3.xlarge`, karena setiap instans `m3.xlarge` setara dengan 8 unit/jam yang dinormalisasi. Penggunaan `m3.xlarge` yang tersisa dikenai biaya dengan tarif Sesuai Permintaan.

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 Instans Sesuai Permintaan berikut di akun A (akun pembelian):

- 2 `m4.xlarge` Linux, instans tenancy default di Availability Zone `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 Instans Sesuai Permintaan 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 Instans Sesuai Permintaan berikut di akun A (akun pembelian):

- 1 x `m4.xlarge` Linux, instans penghunian default di Zona Ketersediaan `us-east-1a`

Seorang pelanggan menjalankan Instans Sesuai Permintaan berikut di akun B tertaut:

- 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 Terpesan regional yang dimiliki oleh akun A diterapkan pada penggunaan di akun A terlebih dahulu, Instans Terpesan zonal yang dimiliki oleh akun C tetap tidak digunakan dan penggunaan di akun B dikenai biaya dengan tarif Sesuai Permintaan.

Untuk informasi selengkapnya, lihat [Instans Terpesan dalam Laporan Manajemen Penagihan dan Biaya](#).

Note

Instans Terpesan Zona mereservasi kapasitas untuk akun pemilik saja dan tidak dapat dibagikan dengan Akun AWS lain. Jika Anda perlu berbagi kapasitas dengan yang lain Akun AWS, gunakan [Reservasi Kapasitas Sesuai Permintaan](#).

Menggunakan Instans Terpesan Anda

Instans Terpesan secara otomatis diterapkan untuk menjalankan Instans Sesuai Permintaan asalkan spesifikasinya cocok. Jika Anda tidak memiliki Instans Sesuai Permintaan yang berjalan yang sesuai dengan spesifikasi Instans Terpesan Anda, Instans Terpesan tidak akan digunakan hingga Anda meluncurkan instans dengan spesifikasi yang diperlukan.

Jika Anda meluncurkan Instans Sesuai Permintaan untuk memanfaatkan keuntungan penagihan Instans Terpesan, pastikan Anda menentukan informasi berikut saat mengonfigurasi Instans Sesuai Permintaan:

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](#).

Untuk contoh tentang bagaimana Instans Terpesan diterapkan ke Instans Sesuai Permintaan berjalan Anda, lihat [Bagaimana Instans Terpesan 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 Instans Sesuai Permintaan yang menggunakan diskon Instans Terpesan Anda. Untuk informasi selengkapnya tentang berbagai metode peluncuran, lihat [Luncurkan instans Anda](#). Anda juga dapat menggunakan Amazon EC2 Auto Scaling untuk meluncurkan instans. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon EC2 Auto Scaling](#).

Bagaimana Anda ditagih

Semua Instans Terpesan menyediakan diskon dibandingkan dengan harga Sesuai Permintaan. 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 Terpesan kedaluwarsa, Anda akan dikenai tarif Sesuai Permintaan 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](#).

Tingkat AWS Gratis tersedia untuk AWS akun baru. Jika menggunakan AWS Tingkat Gratis untuk menjalankan instans Amazon EC2, dan Anda membeli Instans Terpesan, Anda akan dikenai biaya berdasarkan pedoman 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 [Siklus hidup instans](#).

Keuntungan penagihan Instans Terpesan dapat diterapkan ke instans yang berjalan dengan basis per detik. Penagihan per detik tersedia untuk instans yang menggunakan distribusi Linux sumber terbuka, seperti Amazon Linux dan Ubuntu. Penagihan per jam 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 banyak instans secara bersamaan, tetapi hanya dapat menerima keuntungan diskon Instans Terpesan dengan total 3.600 detik per jam. Penggunaan instans yang melebihi 3.600 detik dalam satu jam akan ditagih dengan tarif Sesuai Permintaan.

Misalnya, jika Anda membeli satu Instans Terpesan `m4.xlarge` dan menjalankan empat `m4.xlarge` Instans secara bersamaan selama satu jam, satu instans dikenai biaya pada satu jam

penggunaan Instans Terpesan dan tiga instans lainnya dikenai biaya pada tiga jam penggunaan Sesuai Permintaan.

Namun, jika Anda membeli satu Instans Terpesan `m4.xlarge` dan menjalankan empat instans `m4.xlarge` 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 Terpesan dan 0 jam penggunaan Sesuai Permintaan.

	1:00	1:15	1:30	1:45
Instance 1				
Instance 2				
Instance 3				
Instance 4				

Jika banyak instans yang memenuhi syarat berjalan secara bersamaan, keuntungan penagihan Instans Terpesan diterapkan ke semua instans pada waktu yang sama hingga maksimum 3.600 detik dalam satu jam. Setelah itu, tarif Sesuai Permintaan berlaku.

	1:00	1:15	1:30	1:45
Instance 1				
Instance 2				
Instance 3				
Instance 4				

Uses Reserved Instance Rate for first 3600 seconds of use
Uses On-Demand Rate

Cost Explorer di konsol [Manajemen Penagihan dan Biaya](#) memungkinkan Anda menganalisis penghematan terhadap Instans Sesuai Permintaan yang berjalan. [FAQ Instans Terpesan](#) menyertakan contoh penghitungan nilai daftar.

Jika Anda menutup AWS akun, penagihan On-Demand 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 Billing and Cost Management](#) 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 penggunaan Instans Cadangan menggunakan Laporan AWS Biaya dan Penggunaan. Untuk informasi selengkapnya, lihat [Instans Terpesan](#) di bagian Laporan Biaya dan Penggunaan dalam Panduan Pengguna AWS Billing .

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 yang lain Akun AWS, gunakan [Reservasi Kapasitas Sesuai Permintaan](#).

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 Terpesan Windows dengan SQL Server Standard, SQL Server Web, dan SQL Server Enterprise.
- 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)
```

New console

Untuk melihat nilai harga tetap untuk Instans Terpesan 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, nyalakan Harga di muka, lalu pilih Konfirmasi.

Old console

Untuk melihat nilai harga tetap untuk Instans Terpesan 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 Depan, pilih pengaturan  di sudut kanan atas, pilih Harga Depan, dan pilih Tutup.

Untuk melihat nilai harga tetap untuk Instans Terpesan menggunakan baris perintah

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)
- [DescribeReservedContoh](#) (Amazon EC2 API)

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 [Membeli Instans Terpesan](#).

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](#).

Membeli Instans Terpesan

Untuk membeli Instans Cadangan, cari penawaran Instans Cadangan dari AWS dan penjual pihak ketiga, sesuaikan parameter penelusuran hingga Anda menemukan kecocokan persis yang Anda cari.

Saat mencari Instans Terpesan untuk dibeli, Anda menerima penawaran kuota dari penawaran yang ditampilkan. Saat Anda melanjutkan pembelian, AWS secara otomatis menempatkan batas harga 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 EC2, jika ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga di muka yang lebih rendah, AWS menjual penawaran dengan harga di muka 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.

Untuk membeli dan memodifikasi Instans Terpesan, pastikan bahwa pengguna Anda memiliki izin yang sesuai, seperti kemampuan untuk menjelaskan Zona Ktersediaan. Untuk selengkapnya, lihat [the section called “Cara Menggunakan Instans Cadangan” \(API\)](#) atau [the section called “Cara Menggunakan Instans Cadangan” \(konsol\)](#).

Topik

- [Memilih platform](#)
- [Mengantrekan pembelian Anda](#)
- [Membeli Instans Terpesan Standar](#)
- [Membeli Instans Terpesan Konvertibel](#)
- [Membeli dari Marketplace Instans Terpesan](#)
- [Melihat Instans Terpesan Anda](#)
- [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

Saat membeli sebuah Instans Terpesan. Anda harus memilih penawaran untuk platform yang mewakili sistem operasi untuk instans Anda.

Instans Linux

- Untuk distribusi SUSE Linux dan RHEL, Anda harus memilih penawaran untuk platform spesifik tersebut. Misalnya, untuk platform SUSE Linux atau Red Hat Enterprise Linux.
- Untuk semua distribusi Linux lainnya (termasuk Ubuntu), pilih penawaran untuk platform Linux/UNIX.
- Jika membawa langganan RHEL yang ada, Anda harus memilih penawaran untuk platform Linux/UNIX, bukan penawaran untuk platform Linux Red Hat Enterprise.

Instans Windows

- Untuk Windows dengan SQL Standard, Windows dengan SQL Server Enterprise, dan Windows dengan SQL Server Web, Anda harus memilih penawaran untuk platform spesifik tersebut.
- Untuk semua versi Windows lainnya, pilih penawaran untuk platform Windows.

Note

Ubuntu Pro tidak tersedia sebagai Instans Terpesan. Untuk penghematan yang signifikan dibandingkan dengan harga Instans Sesuai Permintaan, sebaiknya Anda menggunakan Ubuntu Pro dengan Savings Plans. Untuk informasi selengkapnya, lihat [Panduan Pengguna Savings Plans](#).

Important

Jika Anda berencana membeli Instans Terpesan untuk diterapkan ke Instans Sesuai Permintaan yang diluncurkan dari AMI AWS Marketplace, periksa terlebih dahulu bidang PlatformDetails dari AMI tersebut. Bidang PlatformDetails menunjukkan Instans Terpesan yang akan dibeli. Detail platform AMI harus cocok dengan platform Instans Terpesan. Jika tidak, Instans Terpesan tidak akan diterapkan ke Instans Sesuai Permintaan. Untuk informasi tentang cara melihat detail platform AMI, lihat [Memahami informasi penagihan AMI](#).

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 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.

New console

Untuk membeli Instans Terpesan Standar menggunakan konsol

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 Instans 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 lebih rendah, AWS menjual penawaran kepada Anda 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.

 Note

Jika statusnya masuk ke Retired, AWS mungkin belum menerima pembayaran Anda.

Old console

Untuk membeli Instans Terpesan Standar menggunakan konsol

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 Instans Terpesan Standar.
4. Untuk membeli reservasi kapasitas, pilih Hanya tampilkan penawaran yang memiliki kapasitas di sudut kanan atas layar pembelian. Untuk membeli Instans Terpesan wilayah, kosongkan kotak centang.
5. Pilih konfigurasi lain sesuai kebutuhan dan pilih Cari.

Untuk membeli Instans Terpesan Standar dari Marketplace Instans Terpesan, cari Pihak ke-3 di kolom Penjual pada hasil pencarian. Kolom Istilah menampilkan istilah nonstandar.

6. Untuk setiap Instans Terpesan yang ingin Anda beli, masukkan jumlahnya, dan pilih Tambahkan ke Keranjang.
7. Untuk melihat ringkasan Instans Terpesan yang Anda pilih, klik Lihat Keranjang.
8. Jika Pesanan Pada adalah Sekarang, pembelian segera diselesaikan. 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.

Jika, pada saat melakukan pemesanan, ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga lebih rendah, AWS menjual penawaran kepada Anda 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.

Note

Jika statusnya masuk ke `retired`, AWS mungkin belum menerima pembayaran Anda.

Untuk membeli Instans Cadangan Standar menggunakan AWS CLI

1. Temukan Instans Terpesan yang tersedia menggunakan perintah [describe-reserved-instances-offerings](#). Tetapkan `standard` untuk parameter `--offering-class` agar hanya menampilkan Instans Terpesan Standar. Anda dapat menerapkan parameter tambahan untuk mempersempit hasil Anda. Misalnya, jika Anda ingin membeli Instans Terpesan regional `t2.large` dengan penghunian default untuk 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"
```

- 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 mengantrekan pembelian, tambahkan parameter berikut ke panggilan sebelumnya.

```
--purchase-time "2020-12-01T00:00:00Z"
```

- Gunakan perintah [describe-reserved-instances](#) untuk mendapatkan status Instans Cadangan Anda.

```
aws ec2 describe-reserved-instances
```

Atau, gunakan AWS Tools for Windows PowerShell perintah berikut:

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

Setelah pembelian selesai, 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 Instans Terpesan diterapkan](#).

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.

New console

Untuk membeli Instans Terpesan Konvertibel menggunakan konsol

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 lebih rendah, AWS menjual penawaran kepada Anda 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.

Note

Jika statusnya masuk keRetired, AWS mungkin belum menerima pembayaran Anda.

Old console

Untuk membeli Instans Terpesan Konvertibel menggunakan konsol

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, pilih Hanya tampilkan penawaran yang memiliki kapasitas di sudut kanan atas layar pembelian. Untuk membeli Instans Terpesan wilayah, kosongkan kotak centang.
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 segera diselesaikan. 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.

Jika, pada saat melakukan pemesanan, ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga lebih rendah, AWS menjual penawaran kepada Anda 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.

Note

Jika statusnya masuk ke `tired`, AWS mungkin belum menerima pembayaran Anda.

Untuk membeli Instans Cadangan Konvertibel menggunakan AWS CLI

1. Temukan Instans Terpesan yang tersedia menggunakan perintah [describe-reserved-instances-offerings](#). Tentukan `convertible` untuk parameter `--offering-class` agar hanya menampilkan Instans Terpesan Konvertibel. Anda dapat menerapkan parameter tambahan untuk mempersempit hasil. Misalnya, jika Anda ingin membeli Instans Terpesan regional `t2.large` dengan penghunian 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 mengantrekan pembelian, tambahkan parameter berikut ke panggilan sebelumnya.

```
--purchase-time "2020-12-01T00:00:00Z"
```

- Gunakan perintah [describe-reserved-instances](#) untuk mendapatkan status Instans Cadangan Anda.

```
aws ec2 describe-reserved-instances
```

Atau, gunakan AWS Tools for Windows PowerShell perintah berikut:

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

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 Instans Terpesan diterapkan](#).

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. Ketentuan standar penuh dari AWS berjalan selama satu tahun atau tiga tahun.
- Harga di muka** – Instans Terpesan pihak ketiga dapat dijual dengan harga di muka yang berbeda. Biaya penggunaan atau berulang tetap sama dengan biaya yang ditetapkan saat Instans Cadangan awalnya dibeli. AWS

- Tipe Instans Terpesan – Hanya Instans Terpesan Standar Amazon EC2 yang dapat dibeli dari Marketplace Instans Terpesan. Instans Cadangan Konvertibel, Amazon RDS, dan Instans ElastiCache 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 penjual dengan alamat email Anda, sehingga mereka dapat menghubungi Anda mengenai pertanyaan yang terkait dengan penjualan (misalnya, pertanyaan pajak).

Untuk alasan yang sama, AWS bagikan nama badan hukum penjual pada faktur pembelian pembeli. Jika Anda memerlukan informasi tambahan tentang penjual untuk pajak atau alasan terkait, hubungi [AWS Support](#).

Melihat Instans Terpesan Anda

Anda dapat melihat Instans Terpesan yang telah dibeli menggunakan konsol Amazon EC2, atau alat baris perintah.

Untuk melihat Instans Terpesan Anda di konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans Terpesan.
3. Instans Terpesan Anda yang antre, aktif, dan sudah pensiun ditampilkan. Kolom Status menampilkan status.
4. Jika Anda adalah penjual di Marketplace Instans Terpesan, tab Daftar Saya menampilkan status reservasi yang terdaftar di [Marketplace Instans Terpesan](#). Untuk informasi selengkapnya, lihat [Status iklan Instans Terpesan](#).

Untuk melihat Instans Terpesan Anda menggunakan baris perintah

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#) (Alat untuk Windows PowerShell)

Membatalkan antrean pembelian

Anda dapat mengantrekan pembelian hingga tiga tahun ke depan. Anda dapat membatalkan antrean pembelian kapan saja sebelum waktu yang dijadwalkan.

New console

Untuk membatalkan antrean pembelian

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.

Old console

Untuk membatalkan antrean pembelian

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, pilih Ya, Hapus.

Untuk membatalkan antrean pembelian menggunakan baris perintah

- [delete-queued-reserved-instances](#) (AWS CLI)
- [Remove-EC2QueuedReservedInstance](#) (Alat untuk Windows PowerShell)

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.

New console

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.

Old console

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.

Menjual di Marketplace Instans Terpesan

Marketplace Instans Cadangan adalah platform yang mendukung penjualan Instans Cadangan Standar pihak ketiga dan AWS pelanggan yang tidak digunakan, yang bervariasi dalam jangka waktu dan opsi harga. Misalnya, Anda mungkin ingin menjual Instans Cadangan setelah memindahkan instans ke AWS Wilayah baru, mengubah ke jenis instans baru, mengakhiri proyek sebelum jangka waktu kedaluwarsa, saat bisnis Anda perlu diubah, atau jika Anda memiliki kapasitas yang tidak dibutuhkan.

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 EC2, AWS pertama-tama jual Instans Cadangan dengan harga dimuka terendah dalam pengelompokan yang ditentukan. Kemudian, AWS jual Instans Cadangan

dengan harga terendah berikutnya, sampai 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, AWS menagih harga Sesuai Permintaan mulai dari saat Instans Cadangan Anda terjual.

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 Terpesan Konvertibel Amazon EC2 tidak dapat dijual di Marketplace Instans Terpesan.
- 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 Terpesan Standar di Wilayah yang [dininaktifkan 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 Terpesan, instans tersebut hanya dapat dijual setelah AWS menerima pembayaran di muka.
- 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 membebaskan biaya layanan sebesar 12 persen dari total harga dimuka 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 menjual Instans Cadangan di Marketplace Instans Cadangan meskipun mereka memiliki rekening bank AS. Untuk informasi selengkapnya, lihat [Apa perbedaan antara akun AWS India Akun AWS dan Indonesia?](#)

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 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 Anda. 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 menggunakan rekening bank perusahaan, Anda akan diminta untuk mengirimkan informasi tentang rekening bank tersebut 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 tersebut 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 nasihat 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 selengkapnya tentang persyaratan IRS dan Formulir 1099-K, lihat 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 di muka adalah harga satu kali yang dibayar pembeli saat mereka membeli Instans Terpesan.

Karena nilai Instans Cadangan menurun dari waktu ke waktu, secara default, AWS dapat menetapkan harga untuk menurun dalam 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. Batas tersebut bukan batas tahunan.
 - Anda dapat menjual hingga 50.000 USD dalam Instans Terpesan.
 - Anda dapat menjual hingga 5.000 USD Instans Terpesan.

Batasan ini biasanya tidak dapat ditingkatkan, tetapi akan dievaluasi case-by-case berdasarkan permintaan. Untuk meminta kenaikan batas, lengkapi formulir [Kenaikan batas layanan](#). Untuk Tipe batas, pilih Penjualan Instans Terpesan EC2.

- **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 iklan 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 instans 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.

Untuk mencantumkan Instans Cadangan di Marketplace Instans Cadangan menggunakan AWS Management Console

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 Konfigurasi 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 iklan Anda di konsol

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.

Untuk mengelola Instans Cadangan di Marketplace Instans Cadangan menggunakan AWS CLI

1. Dapatkan daftar Instans Terpesan Anda dengan menggunakan perintah [describe-reserved-instances](#).
2. Catat ID Instans Terpesan yang ingin Anda iklankan dan panggil [create-reserved-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-reserved-instances-listings](#).

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, AWS menghentikan Instans Cadangan 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 parsial ini, AWS membuat reservasi baru dengan jumlah tiga untuk menunjukkan sisa reservasi yang masih untuk 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.

Untuk melacak status daftar Instans Terpesan di konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di halaman navigasi, pilih Instans Terpesan.
3. Pilih tab Iklan Saya.

Tab Iklan Saya berisi nilai Status Iklan. Tab ini juga berisi informasi tentang jangka waktu, harga iklan, dan perincian jumlah instans dalam daftar yang tersedia, tertunda, dijual, dan dibatalkan.

Anda juga dapat menggunakan perintah [describe-reserved-instances-listings](#) dengan filter yang sesuai untuk mendapatkan informasi tentang iklan Anda.

Mendapatkan pembayaran

Segera setelah AWS menerima dana dari pembeli, pesan dikirim ke email akun pemilik terdaftar untuk Instans Cadangan yang dijual.

AWS mengirimkan transfer kawat 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 sekali sehari. Anda akan menerima email dengan laporan pencairan setelah dana dikeluarkan. Ingatlah bahwa Anda tidak dapat menerima pencairan hingga 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 kawat langsung ke rekening bank Anda. AWS membebankan biaya layanan sebesar 12 persen dari total harga dimuka 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 menelepon AWS Support karena pembeli perlu menghubungi Anda untuk faktur atau untuk alasan terkait pajak lainnya, AWS mungkin perlu memberikan alamat email 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 saran pajak, tetapi jika spesialis pajak Anda menentukan bahwa Anda memerlukan informasi tambahan spesifik, [hubungi AWS Support](#).

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 cocok dengan parameter baru akan dikenai tarif Sesuai Permintaan, 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	<p>Instans Terpesan zonal tercakup dalam Zona Ketersediaan dan kapasitas terpesan di Zona Ketersediaan tersebut. Jika Anda mengubah cakupan dari Zona Ketersediaan ke Wilayah (dengan kata lain, dari zonal ke regional), Anda kehilangan keuntungan reservasi kapasitas.</p> <p>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 penggunaan 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 fleksibilitas Zona Ketersediaan dan fleksibilitas ukuran instans (jika berlaku).</p>

Atribut yang dapat dimodifikasi	Platform yang didukung	Batasan dan pertimbangan
		Untuk informasi selengkapnya, lihat Bagaimana Instans Terpesan diterapkan .
Ubah ukuran instans dalam keluarga dan generasi instans yang sama	Linux/UNIX saja 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
- Instans Cadangan bukan instance G4, G4ad, G4dn, G5, G5g, Inf1, atau Inf2.

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 [generasi](#) 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 tidak dapat mengubah ukuran Instans Terpesan untuk instans berikut karena masing-masing hanya memiliki satu ukuran:
 - `t1.micro`
- Anda tidak dapat mengubah ukuran instans Instans Cadangan untuk kombinasi keluarga instans, generasi, dan atribut berikut:
 - G4ad
 - G4dn
 - G5
 - G5g
 - Inf1
 - Inf2
- 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.

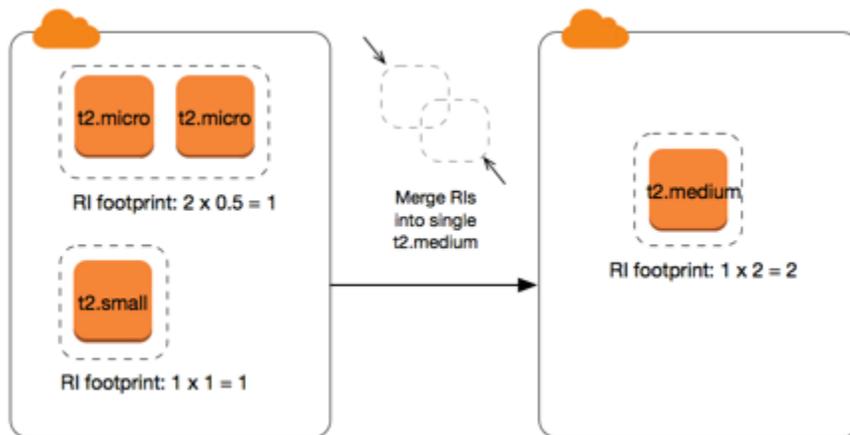
Untuk menghitung jejak ukuran instans dari Instans Terpesan, kalikan jumlah instans dengan faktor normalisasi. Di konsol Amazon EC2, faktor normalisasi diukur dalam satuan. Tabel berikut menjelaskan faktor normalisasi untuk ukuran instans dalam suatu keluarga instans. Misalnya, `t2.medium` memiliki faktor 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

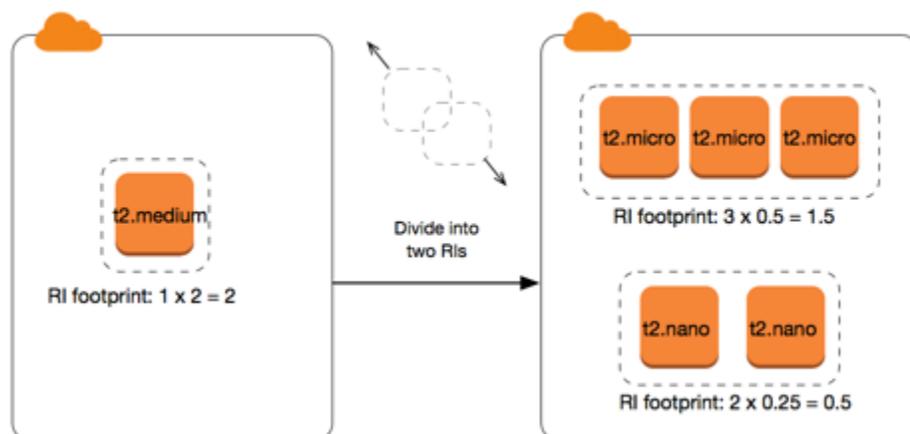
Ukuran instans	Faktor normalisasi
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256
48xlarge	384
56xlarge	448
112xlarge	896

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
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-*.metal	896

Misalnya, instans `i3.metal` memiliki faktor normalisasi 128. Jika membeli Instans Terpesan Amazon Linux/Unix penghunian default `i3.metal`, Anda dapat membagi reservasi sebagai berikut:

- Sebuah instans `i3.16xlarge` berukuran sama dengan instans `i3.metal`, sehingga faktor normalisasinya adalah 128 ($128/1$). Reservasi untuk satu instans `i3.metal` dapat dimodifikasi menjadi satu instans `i3.16xlarge`.
- Sebuah instans `i3.8xlarge` berukuran setengah dari instans `i3.metal`, sehingga faktor normalisasinya adalah 64 ($128/2$). Reservasi untuk satu instans `i3.metal` dapat dibagi menjadi dua instans `i3.8xlarge`.
- Sebuah instans `i3.4xlarge` berukuran seperempat dari instans `i3.metal`, sehingga 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 footprint](#) dari reservasi asli yang ingin Anda modifikasi dan pastikan bahwa itu cocok dengan total ukuran instance footprint konfigurasi baru Anda.

New console

Untuk mengubah Instans Cadangan Anda menggunakan AWS Management Console

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.

Note

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 instance 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

Old console

Untuk mengubah Instans Cadangan Anda menggunakan AWS Management Console

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.

Note

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, lalu pilih Lanjutkan:
 - 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 Instance:** 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. Untuk mengonfirmasi pilihan modifikasi setelah Anda selesai menentukan konfigurasi target Anda, pilih Kirim Modifikasi.
 5. 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

Untuk memodifikasi Instans Terpesan Anda menggunakan baris perintah

1. Untuk memodifikasi Instans Terpesan, Anda dapat menggunakan salah satu perintah berikut:
 - [modify-reserved-instances](#) (AWS CLI)
 - [Edit-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)
2. Untuk mendapatkan status permintaan modifikasi Anda (`processing`, `fulfilled`, atau `failed`), gunakan salah satu perintah berikut:
 - [describe-reserved-instances-modifications](#) (AWS CLI)
 - [Get-EC2ReservedInstancesModification](#) (AWS Tools for Windows PowerShell)

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 Terpesan Konvertibel hanya dapat ditukar dengan Instans Terpesan Konvertibel lain yang saat ini ditawarkan oleh AWS.
- Instans Terpesan Konvertibel dikaitkan dengan Wilayah tertentu, yang ditetapkan selama jangka waktu reservasi. Anda tidak dapat menukar Instans Terpesan Konvertibel dengan Instans Terpesan Konvertibel di Wilayah lain.
- 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 di muka yang diperlukan untuk pertukaran (biaya true-up) kurang dari 0,00 USD, AWS secara otomatis memberi Anda jumlah instans dalam Instans Terpesan Konvertibel yang memastikan bahwa biaya true-up adalah 0,00 USD atau lebih.

Note

Jika nilai total (harga dimuka + harga per jam * jumlah jam yang tersisa) dari Instans Cadangan Konvertibel baru kurang dari nilai total Instans Cadangan Konvertibel yang dipertukarkan, AWS secara otomatis memberi Anda sejumlah instans dalam Instans Cadangan Konvertibel yang memastikan bahwa nilai totalnya sama atau lebih tinggi dari Instans Cadangan Konvertibel yang dipertukarkan.

- 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 Instans Terpesan Konvertibel Pembayaran Semua di Muka dan Sebagian di Muka dengan Instans Terpesan Konvertibel Tanpa Pembayaran 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 Cadangan Konvertibel baru kurang dari nilai total Instans Cadangan Konvertibel yang dipertukarkan, AWS secara otomatis memberi Anda sejumlah instans dalam Instans Cadangan Konvertibel yang memastikan bahwa nilai totalnya sama atau lebih tinggi dari Instans Cadangan Konvertibel yang dipertukarkan.

- 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 Terpesan Konvertibel yang Anda inginkan untuk menentukan banyaknya reservasi instans yang dapat Anda terima dari pertukaran tersebut.

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:

$$3.5 = 3 \text{ whole Convertible Reserved Instances} + 1 \text{ additional Convertible Reserved Instance}$$

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.

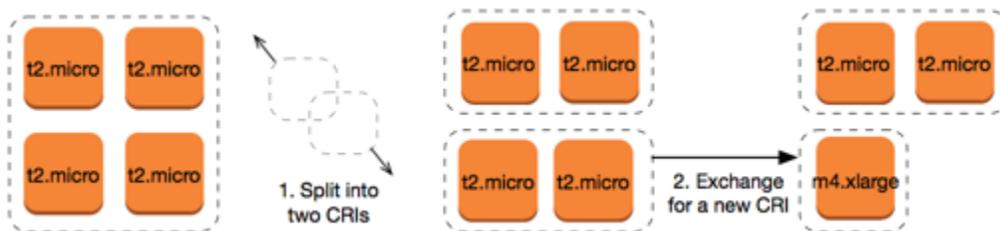
Menukar sebagian dari Instans Terpesan Konvertibel

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`:

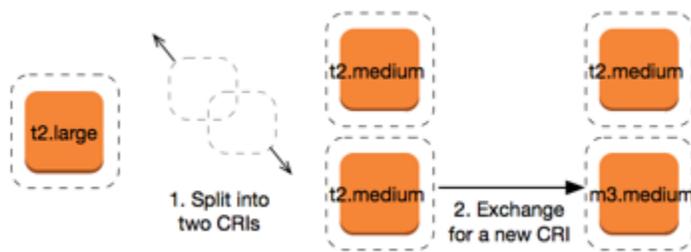
1. Modifikasi Instans Terpesan Konvertibel `t2.micro` dengan membaginya menjadi dua Instans Terpesan Konvertibel `t2.micro`, masing-masing dua instans.
2. Tukar salah satu Instans Terpesan Konvertibel `t2.micro` yang baru dengan Instans Terpesan Konvertibel `m4.xlarge`.



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 lebih lanjut, lihat [Dukungan untuk memodifikasi ukuran instans](#) dan [Mengirimkan permintaan pertukaran](#)

Mengirimkan permintaan pertukaran

Anda dapat menukar Instans Terpesan Konvertibel menggunakan konsol Amazon EC2 atau alat baris perintah.

Menukar Instans Terpesan Konvertibel menggunakan konsol

Anda dapat mencari penawaran Instans Terpesan Konvertibel dan memilih konfigurasi baru Anda dari pilihan yang disediakan.

New console

Untuk menukar Instans Terpesan Konvertibel menggunakan konsol Amazon EC2

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.

Old console

Untuk menukar Instans Terpesan Konvertibel menggunakan konsol Amazon EC2

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. Kolom Jumlah Instans menampilkan jumlah Instans Terpesan yang Anda terima untuk pertukaran tersebut. Jika Anda telah memilih Instans Terpesan Konvertibel yang memenuhi kebutuhan Anda, pilih Tukar.

Instans Terpesan yang ditukar akan dipensiunkan, dan Instans Terpesan baru ditampilkan di konsol Amazon EC2. Proses ini memerlukan waktu beberapa menit untuk diterapkan.

Menukar Instans Terpesan Konvertibel menggunakan antarmuka baris perintah

Untuk menukar Instans Terpesan Konvertibel, temukan terlebih dahulu Instans Terpesan Konvertibel baru yang memenuhi kebutuhan Anda:

- [describe-reserved-instances-offerings](#) (AWS CLI)
- [Get-EC2ReservedInstancesOffering](#)(Alat untuk Windows PowerShell)

Dapatkan penawaran untuk pertukaran, yang mencakup jumlah Instans Terpesan yang Anda dapatkan dari pertukaran, dan biaya aktual untuk pertukaran:

- [get-reserved-instances-exchange-quote](#) (AWS CLI)
- [GetEC2-ReservedInstancesExchangeQuote](#) (Alat untuk Windows) PowerShell

Terakhir, lakukan pertukaran:

- [accept-reserved-instances-exchange-quote](#) (AWS CLI)
- [Confirm-EC2ReservedInstancesExchangeQuote](#)(Alat untuk Windows PowerShell)

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)

Contoh di `running` negara bagian dihitung terhadap kuota Anda. Contoh yang ada di `pending`, `stopping`, `stopped`, dan `hibernated` negara bagian 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.

EC2 > Reserved Instances

Reserved Instances (32) [Info](#)

Filter by attributes or search by keyword

Instance ty...	Scope	Availabilit...	Instance count	Start	Expires	Offering cl...
<input checked="" type="checkbox"/> t3.micro	Region	-	10	August 27, 2022, 15:29 (UTC+2:00)	August 27, 2023, 15:29 (UTC+2:00)	Standard
<input type="checkbox"/> t3.micro	Region	-	4	November 8, 2021, 14:19 (UTC+2:00)	November 8, 2022, 14:19 (UTC+2:00)	Standard

1 Reserved Instance selected

Details | My Listings

Reserved Instance ID: 2fbf16dd-98b6-4a3a-955f-83f87790f04b [Info](#)

Instance type t3.micro	Scope Region	Instance count 10	Availability Zone -
Start August 27, 2022, 15:29 (UTC+2:00)	Platform Linux/UNIX	Expires August 27, 2023, 15:29 (UTC+2:00)	Term 1 year
Payment option All upfront	Time left around 50 weeks 6 days	Upfront price \$59.00	Offering class Standard
Usage price \$0.00	State Active	Hourly charges \$0.00	Tenancy Default

AWS CLI

Untuk melihat jumlah Instans Terpesan yang telah Anda beli

Gunakan perintah CLI [describe-reserved-instances](#) dan tentukan ID konfigurasi Instans Terpesan.

```
aws ec2 describe-reserved-instances \
  --reserved-instances-ids 2fbf16dd-98b6-4a3a-955f-83f87790f04b \
  --output table
```

Output contoh – Bidang InstanceCount menunjukkan bahwa ada 10 Instans Terpesan 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 | 2fbf16dd-98b6-4a3a-955f-83f87790f04b ||
|| Scope               | Region                 ||
|| Start               | 2022-08-27T13:29:45.938000+00:00 ||
|| State               | active                 ||
|| UsagePrice          | 0.0                    ||
|+-----+-----+
|||                               |||
|||             RecurringCharges |||
||+-----+-----+
||| Amount                | 0.0                    |||
||| Frequency             | Hourly                 |||
||+-----+-----+

```

Pertimbangan

Instans Terpesan regional menerapkan diskon ke Instans Sesuai Permintaan yang sedang berjalan. Batas Instans Sesuai Permintaan default adalah 20. Anda tidak dapat melebihi batas Instans Sesuai Permintaan yang sedang berjalan dengan membeli Instans Terpesan regional. Misalnya, jika Anda sudah memiliki 20 Instans Sesuai Permintaan yang sedang berjalan dan membeli 20 Instans Terpesan regional, 20 Instans Terpesan regional digunakan untuk menerapkan diskon ke 20 Instans Sesuai Permintaan yang sedang berjalan. Jika membeli lebih banyak Instans Terpesan regional, Anda tidak akan dapat meluncurkan lebih banyak instans karena Anda telah mencapai batas Instans Sesuai Permintaan.

Sebelum membeli Instans Terpesan regional, pastikan batas Instans Sesuai Permintaan Anda sesuai atau melebihi jumlah Instans Terpesan regional yang ingin dimiliki. Jika perlu, pastikan Anda meminta peningkatan batas Instans Sesuai Permintaan sebelum membeli lebih banyak Instans Terpesan regional.

Instans Terpesan zonal—Instans Terpesanyang dibeli untuk Zona Ketersediaan tertentu—memberikan reservasi kapasitas serta diskon. Anda dapat melebihi batas Instans Sesuai Permintaan yang sedang berjalan dengan membeli Instans Terpesan zonal. Misalnya, jika Anda sudah memiliki 20 Instans Sesuai Permintaan yang sedang berjalan dan membeli 20 Instans Terpesan zonal, Anda dapat meluncurkan 20 Instans Sesuai Permintaan lain yang sesuai dengan spesifikasi Instans Terpesan Anda, sehingga Anda memiliki 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. Untuk informasi selengkapnya, lihat [Melihat kuota Anda saat ini](#) dan [Meminta peningkatan](#).

Instans Spot

Instans Spot adalah instans yang menggunakan kapasitas EC2 tidak terpakai yang tersedia dengan harga yang lebih rendah dari harga Sesuai Permintaan. 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 pembelian instans](#).

Topik

- [Konsep](#)
- [Cara memulai](#)
- [Layanan-layanan terkait](#)
- [Penetapan harga dan penghematan](#)

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 utama antara Instans Spot dan Instans Sesuai Permintaan

Tabel berikut mencantumkan daftar perbedaan utama antara Instans Spot dan [Instans Sesuai Permintaan](#).

	Instans Spot	Instans Sesuai Permintaan
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.
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 Instans Sesuai Permintaan 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 Instans Sesuai Permintaan diinterupsi (dihentikan, hibernasi, atau diakhiri).

	Instans Spot	Instans Sesuai Permintaan
Interupsi instans	Anda dapat menghentikan dan memulai Instans Spot yang didukung Amazon EBS. Selain itu, Amazon EC2 dapat menginterupsi Instans Spot individu jika kapasitas tidak lagi tersedia.	Anda menentukan kapan Instans Sesuai Permintaan diinterupsi (dihentikan, hibernasi, atau diakhiri).

Cara memulai

Hal pertama yang perlu Anda lakukan adalah menyiapkan untuk menggunakan Amazon EC2. Memiliki pengalaman meluncurkan Instans Sesuai Permintaan juga dapat membantu sebelum meluncurkan Instans Spot.

Dasar-dasar Spot

- [Cara kerja Instans Spot](#)

Berkeja dengan Instans Spot

- [Membuat permintaan Instans Spot](#)
- [Dapatkan informasi status permintaan](#)
- [Interupsi Instans Spot](#)

Layanan-layanan terkait

Anda dapat menyediakan Instans Spot secara langsung menggunakan Amazon EC2. Anda juga dapat menyediakan Instans Spot menggunakan layanan lain di AWS. Untuk informasi selengkapnya, lihat dokumentasi berikut.

Amazon EC2 Auto Scaling dan Instans Spot

Anda dapat membuat templat atau konfigurasi peluncuran sehingga Amazon EC2 Auto Scaling dapat meluncurkan Instans Spot. Untuk informasi selengkapnya, lihat [Meminta Instans Spot untuk aplikasi toleransi kesalahan dan fleksibel](#) serta [Grup Auto Scaling dengan banyak tipe instans dan opsi pembelian](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

Amazon EMR dan Instans Spot

Ada skenario yang dapat berguna untuk menjalankan Instans Spot di kluster Amazon EMR. Untuk informasi selengkapnya, lihat [Instans Spot](#) dan [Kapan Anda Harus Menggunakan Instans Spot](#) dalam Panduan Manajemen Amazon EMR.

AWS CloudFormation template

AWS CloudFormation memungkinkan Anda untuk membuat dan mengelola koleksi sumber AWS daya menggunakan template dalam format JSON. Untuk informasi selengkapnya, lihat [Pembaruan Instans Spot EC2 - Auto Scaling CloudFormation](#) dan Integrasi.

AWS SDK for Java

Anda dapat menggunakan bahasa pemrograman Java untuk mengelola Instans Spot Anda. Untuk informasi selengkapnya, lihat [Tutorial: Instans Spot Amazon EC2](#) dan [Tutorial: Pengelolaan Permintaan Spot Amazon EC2 Lanjutan](#).

AWS SDK for .NET

Anda dapat menggunakan lingkungan pemrograman .NET untuk mengelola Instans Spot Anda. Untuk informasi selengkapnya, lihat [Tutorial: Instans Spot Amazon EC2](#).

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 dicakup oleh Savings Plans. Jika Anda memiliki Savings Plan, itu tidak memberikan penghematan tambahan selain tabungan yang sudah Anda dapatkan dari menggunakan Instans Spot. Selain itu, pengeluaran Anda untuk Instans Spot tidak menerapkan komitmen dalam Compute Savings Plans Anda.

Tampilkan harga

Untuk melihat harga Spot terendah saat ini (diperbarui setiap lima menit) per Wilayah AWS 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](#) (AWS CLI). Untuk informasi selengkapnya, lihat [Riwayat harga Instans Spot](#).

Kami secara independen memetakan Availability Zone ke kode untuk masing-masing kode 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 Spot EC2

Instans Spot Amazon EC2 adalah kapasitas komputasi EC2 cadangan AWS Cloud yang tersedia untuk Anda dengan penghematan hingga 90% dibandingkan dengan harga Sesuai Permintaan. Satu-satunya perbedaan antara Instans Sesuai Permintaan dan Instans Spot adalah bahwa Instans Spot dapat diinterupsi oleh Amazon EC2, dengan notifikasi dua menit, ketika Amazon EC2 membutuhkan kapasitas kembali.

Instans Spot direkomendasikan untuk aplikasi tanpa stateless, toleransi kesalahan, dan fleksibel. Misalnya, Instans Spot berfungsi dengan baik untuk big data, beban kerja terkontainer, CI/CD, server web stateless, komputasi performa tinggi (HPC), dan beban kerja rendering.

Saat berjalan, Instans Spot sama persis dengan Instans Sesuai Permintaan. 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 agar fleksibel tentang jenis instans dan Availability Zone memberikan peluang terbaik untuk ketersediaan tinggi, tidak ada jaminan bahwa kapasitas akan tersedia, karena lonjakan permintaan untuk Instans Sesuai Permintaan dapat mengganggu beban kerja pada Instans Spot.

Kami sangat tidak menyarankan menggunakan Instans Spot untuk beban kerja ini atau mencoba gagal ke Instans Sesuai Permintaan untuk menangani interupsi atau periode tidak tersedianya. Kegagalan pada Instans Sesuai Permintaan dapat secara tidak sengaja mendorong interupsi untuk Instans Spot Anda yang lain. Selain itu, jika Instans Spot untuk kombinasi tipe instans dan Availability Zone terputus, mungkin akan sulit bagi Anda untuk mendapatkan Instans Sesuai Permintaan 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](#)
- [Menggunakan grup EC2 Auto Scaling atau EC2 Fleet untuk mengelola kapasitas agregat Anda](#)
- [Menggunakan strategi alokasi harga dan kapasitas yang dioptimalkan](#)
- [Gunakan AWS layanan terintegrasi untuk mengelola Instans Spot](#)
- [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 kembali dan pemberitahuan gangguan, dan kemudian memicu pos pemeriksaan untuk kemajuan beban kerja Anda atau menangani gangguan dengan anggun. 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 hanya dapat menskalakan secara horizontal, Anda harus menyertakan tipe instans generasi sebelumnya karena permintaan dari pelanggan Sesuai Permintaan lebih sedikit.

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.

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 tentang strategi alokasi, lihat [Instans Spot](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling dan [Ketika beban kerja memiliki biaya interupsi yang tinggi](#) dalam panduan pengguna ini.

Gunakan AWS layanan terintegrasi untuk mengelola Instans Spot

AWS Layanan lain terintegrasi dengan Spot untuk mengurangi biaya komputasi secara keseluruhan tanpa perlu mengelola instans atau armada individu. Kami menyarankan Anda mempertimbangkan solusi berikut untuk beban kerja yang berlaku: Amazon EMR, Amazon Elastic Container Service, AWS Batch, Amazon Elastic SageMaker, Kubernetes Service, Amazon, dan Amazon. AWS Elastic

Beanstalk 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	<ul style="list-style-type: none"> Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran. Anda ingin mengotomatiskan manajemen siklus hidup melalui API yang dapat dikonfigurasi. 	Buat grup Auto Scaling yang mengelola siklus hidup instans Anda sambil mempertahankan jumlah instans yang diinginkan. Mendukung penskalaan horizontal (menambahkan lebih banyak instans) antara batas minimum dan maksimum yang ditentukan.	Ya
CreateFleet	<ul style="list-style-type: none"> Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran. Anda ingin mengelola sendiri siklus hidup instans Anda. 	Buat armada Instans Sesuai Permintaan dan Instans Spot dalam satu permintaan dengan banyak spesifikasi peluncuran yang bervariasi menurut tipe instans, AMI, Zona Ketersediaan, atau subnet. Strategi alokasi	Ya – dalam mode instant jika Anda tidak memerlukan penskalaan otomatis

API	Kapan harus menggunakan?	Kasus penggunaan	Haruskah saya menggunakan API ini?
	<ul style="list-style-type: none"> Jika Anda tidak memerlukan penskalaan otomatis, kami sarankan Anda menggunakan armada tipe <code>instant</code>. 	<p>Instans Spot default ke <code>lowest-price</code> per unit, tetapi Anda dapat mengubahnya menjadi <code>price-capacity-optimized</code>, <code>capacity-optimized</code>, atau <code>diversified</code>.</p>	
<p>RunInstances</p>	<ul style="list-style-type: none"> Anda sudah menggunakan <code>RunInstances</code> API untuk meluncurkan Instans Sesuai Permintaan, dan Anda hanya ingin mengubah untuk meluncurkan Instans Spot dengan mengubah satu parameter. Anda tidak memerlukan banyak instans dengan tipe instans yang berbeda. 	<p>Luncurkan sejumlah tertentu instans menggunakan AMI dan satu tipe instans.</p>	<p>Tidak - karena <code>RunInstances</code> tidak mengizinkan jenis instance campuran dalam satu permintaan</p>

API	Kapan harus menggunakan?	Kasus penggunaan	Haruskah saya menggunakan API ini?
RequestSpotArmada	<ul style="list-style-type: none"> • Kami sangat tidak menyarankan menggunakan RequestSpotFleet API karena ini adalah API lama tanpa investasi yang direncanakan. • Jika Anda ingin mengelola siklus hidup instance, gunakan API. CreateFleet • Jika Anda tidak ingin mengelola siklus hidup instance, gunakan API. CreateAutoScalingGroup 	<p>JANGAN GUNAKAN. RequestSpotFleet adalah API lama tanpa investasi yang direncanakan.</p>	<p>Tidak</p>
RequestSpotContoh	<ul style="list-style-type: none"> • Kami sangat tidak menyarankan menggunakan RequestSpotInstances API karena ini adalah API lama tanpa investasi yang direncanakan. 	<p>JANGAN GUNAKAN. RequestSpotInstances adalah API lama tanpa investasi yang direncanakan.</p>	<p>Tidak</p>

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 instans peluncuran](#) di konsol Amazon EC2 atau perintah [AWS CLI run-instances](#). Untuk informasi selengkapnya, lihat [Membuat permintaan Instans Spot](#).
- 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 permintaan Armada Spot](#).

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).

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](#).

Anda dapat menghentikan, memulai, melakukan boot ulang, atau mengakhiri Instans Spot yang didukung Amazon EBS. Layanan Spot dapat menghentikan, mengakhiri, atau menghibernasi Instans Spot saat interupsi.

Daftar Isi

- [Meluncurkan Instans Spot dalam grup peluncuran](#)
- [Meluncurkan Instans Spot dalam grup Zona Ketersediaan](#)
- [Meluncurkan Instans Spot di VPC](#)

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 Anda dengan cara yang sama seperti menentukan subnet untuk Instans Sesuai Permintaan Anda.

- [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.

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.

Saat permintaan Anda dipenuhi, Instans Spot Anda diluncurkan dengan harga Spot saat ini, tidak melebihi harga Sesuai Permintaan. Anda dapat melihat riwayat harga Spot selama 90 hari terakhir, memfilter menurut tipe instans, sistem operasi, dan Zona Ketersediaan.

Untuk melihat harga Spot saat ini

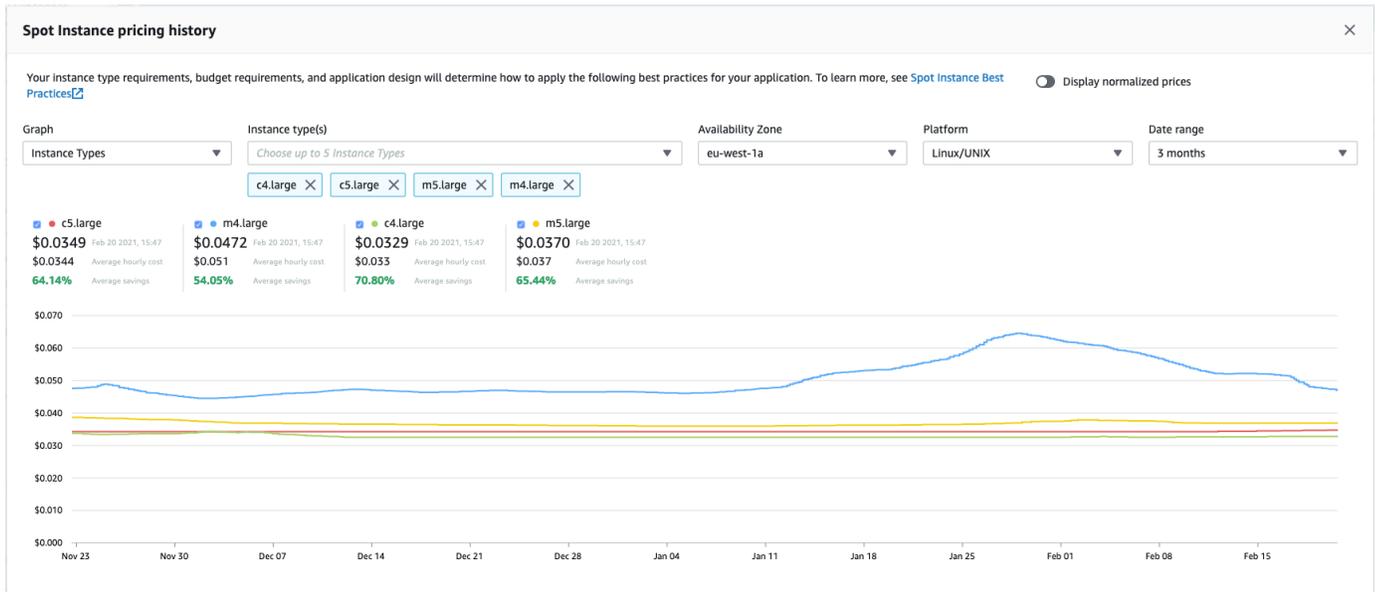
Untuk harga Instans Spot saat ini, lihat [Harga Instans Spot Amazon EC2](#).

Untuk melihat riwayat harga Spot menggunakan konsol

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.

Untuk melihat riwayat harga Spot menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya, lihat [Akses Amazon EC2](#).

- [describe-spot-price-history](#) (AWS CLI)
- [Get-EC2SpotPriceHistory](#) (AWS Tools for Windows PowerShell)

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

Instance Type	vCPU hours	Mem(GiB)-hours	On-Demand total	Savings
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.
- vCPU-jam – 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.
- Total Sesuai Permintaan – Jumlah total yang harus Anda bayarkan untuk kerangka waktu yang dipilih jika Anda meluncurkan instans ini sebagai Instans Sesuai Permintaan.
- Total Spot – Jumlah total yang harus dibayar untuk kerangka waktu yang dipilih.

- Penghematan – Persentase yang Anda hemat dengan tidak membayar harga Sesuai Permintaan.
- Biaya rata-rata per vCPU-jam – Biaya rata-rata per jam dari penggunaan vCPU di semua Instans Spot untuk kerangka waktu yang dipilih, dihitung sebagai berikut: $\text{Biaya rata-rata per vCPU-jam} = \frac{\text{Total Spot}}{\text{vCPU-jam}}$.
- Biaya rata-rata per mem (GiB) -jam - Biaya rata-rata per jam menggunakan GiBs seluruh Instans Spot untuk kerangka waktu yang dipilih, dihitung sebagai berikut: $\text{Biaya rata-rata per mem (GiB) -jam} = \frac{\text{Total spot}}{\text{Mem (GiB) -jam}}$.
- 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.

Untuk melihat informasi penghematan 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 ID permintaan Armada Spot dan gulir ke bagian Penghematan.

Atau, pilih kotak centang di samping ID permintaan Armada Spot, lalu pilih tab Penghematan.

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 menggunakan konsol

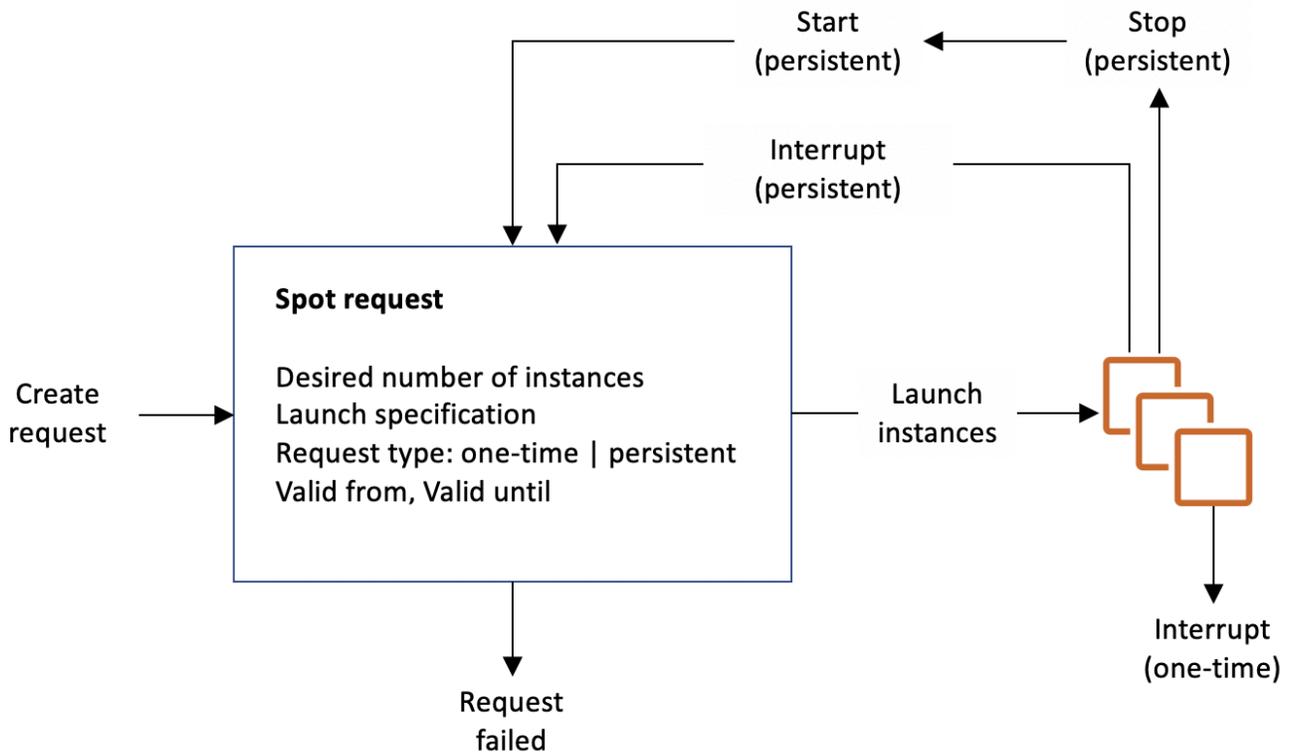
1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih Riwayat Penghematan.

Cara Menggunakan 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.

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

- [Status permintaan Instans Spot](#)
- [Menentukan penghunian untuk Instans Spot Anda](#)
- [Peran terkait layanan untuk permintaan Instans Spot](#)
- [Membuat permintaan Instans Spot](#)
- [Temukan Instans Spot Anda](#)
- [Menandai permintaan Instans Spot](#)
- [Membatalkan permintaan Instans Spot](#)

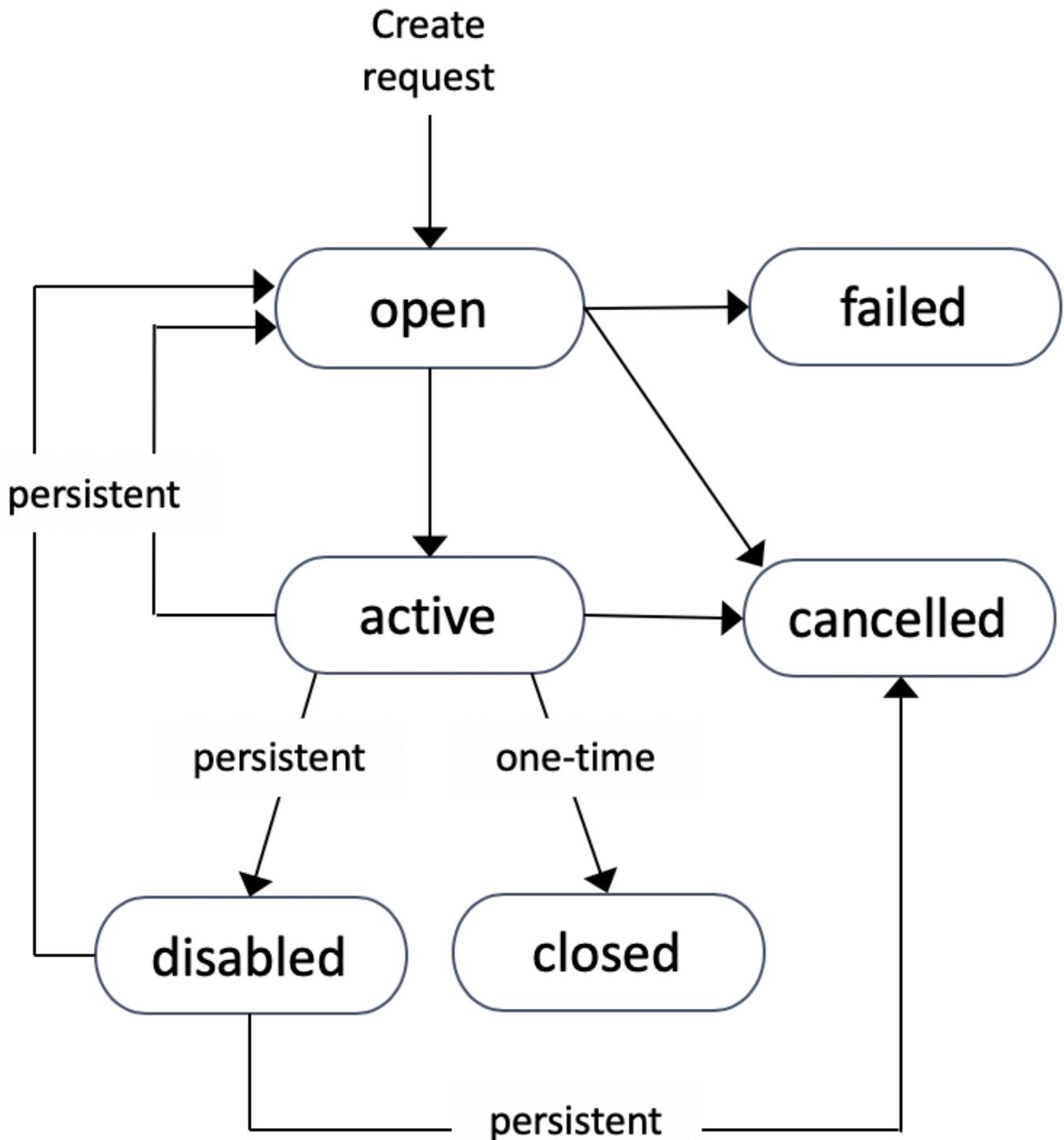
- [Menghentikan Instans Spot](#)
- [Memulai Instans Spot](#)
- [Menghentikan Instans Spot](#)
- [Contoh spesifikasi peluncuran permintaan Instans Spot](#)

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 [Status permintaan spot](#).

Menentukan penghunian untuk Instans Spot Anda

Anda dapat menjalankan Instans Spot pada perangkat keras penghuni tunggal. Instans Spot Khusus secara fisik terisolasi dari instans milik akun lain AWS. Untuk informasi selengkapnya, lihat [Instans Khusus](#) dan halaman produk [Instans 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 [Membuat permintaan Instans Spot](#).
- Minta Instans Spot di VPC dengan penghunian instans dedicated. Untuk informasi selengkapnya, lihat [Buat VPC dengan penghunian instans khusus](#). Anda tidak dapat meminta Instans Spot dengan penghunian default jika Anda memintanya di VPC dengan penghunian instans 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.

Peran tertaut layanan untuk permintaan Instans 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 Peran terkait layanan menyediakan cara aman untuk mendelegasikan izin ke AWS layanan karena hanya layanan tertaut yang dapat mengambil peran terkait layanan. Untuk informasi selengkapnya, lihat [Menggunakan Peran Tertaut Layanan](#) di Panduan Pengguna IAM.

Amazon EC2 menggunakan peran terkait layanan yang diberi nama `AWSServiceRoleForEC2Spot` untuk meluncurkan dan mengelola Instans Spot atas nama Anda.

Izin diberikan oleh `AWSServiceRoleForEC2Spot`

Amazon EC2 menggunakan `AWSServiceRoleForEC2Spot` untuk menyelesaikan tindakan berikut:

- `ec2:DescribeInstances` – Menjelaskan Instans Spot
- `ec2:StopInstances` – Menghentikan Instans Spot
- `ec2:StartInstances` – Memulai Instans Spot

Membuat peran terkait layanan

Dalam sebagian besar situasi, Anda tidak perlu membuat peran terkait layanan secara manual. Amazon EC2 membuat peran `AWSServiceRoleForEC2Spot` terkait layanan saat pertama kali Anda meminta Instans Spot menggunakan konsol.

Jika Anda memiliki permintaan Instans Spot aktif sebelum Oktober 2017, saat Amazon EC2 mulai mendukung peran terkait layanan ini, Amazon EC2 membuat peran tersebut di akun Anda. `AWSServiceRoleForEC2Spot` AWS Untuk informasi selengkapnya, lihat [Peran Baru Muncul di Akun Saya](#) dalam Panduan Pengguna IAM.

Jika Anda menggunakan AWS CLI atau API untuk meminta Instance 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 Berikutnya: Tinjau.
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 perlu lagi menggunakan Instans Spot, sebaiknya hapus `AWSServiceRoleForEC2Spot` peran tersebut. 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 terkelola pelanggan untuk enkripsi, Anda harus memberikan izin peran untuk menggunakan kunci `AWSServiceRoleForEC2Spot` 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 lebih lanjut, lihat [Menggunakan hibah](#) dan [Menggunakan kebijakan kunci di AWS KMS](#) di Panduan Developer AWS Key Management Service .

Untuk memberikan izin peran `AWSServiceRoleForEC2Spot` untuk menggunakan kunci terkelola pelanggan

- Gunakan perintah [create-grant](#) untuk menambahkan hibah ke kunci yang dikelola pelanggan dan untuk menentukan prinsipal (peran `AWSServiceRoleForEC2Spot` terkait layanan) yang diberikan izin untuk melakukan operasi yang diizinkan hibah. Kunci yang dikelola pelanggan ditentukan oleh parameter `key-id` dan ARN kunci yang dikelola pelanggan. Prinsipal ditentukan oleh `grantee-principal` parameter dan ARN dari peran terkait `AWSServiceRoleForEC2Spot` 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"
```

Membuat permintaan Instans Spot

Anda dapat menggunakan [wizard instans peluncuran](#) di konsol Amazon EC2 atau AWS CLI perintah [run-instances untuk meminta Instans](#) Spot dengan cara yang sama seperti Anda dapat meluncurkan Instans Sesuai Permintaan. Metode ini hanya direkomendasikan karena alasan berikut:

- Anda telah menggunakan [wizard peluncuran instans](#) atau perintah [run-instances](#) untuk meluncurkan Instans Sesuai Permintaan, dan hanya ingin mengubah untuk 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 banyak tipe instans, serta tidak dapat meluncurkan Instans Spot dan Instans Sesuai Permintaan dalam permintaan yang sama. Untuk metode yang lebih disukai untuk meluncurkan Instans Spot, yang mencakup peluncuran armada yang menyertakan Instans Spot dan Instans Sesuai Permintaan dengan banyak tipe 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 [Status permintaan spot](#).

New console

Untuk membuat permintaan Instans Spot menggunakan wizard peluncuran instans

Langkah 1–9 adalah langkah yang sama yang akan Anda gunakan untuk meluncurkan Instans Sesuai Permintaan. 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 [Jenis 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 Konfigurasi 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, konfigurasi permintaan Instans Spot sebagai berikut:

- a. Pada bagian Opsi Pembelian, 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 dengan harga Sesuai Permintaan, atau Anda dapat menentukan jumlah maksimum yang bersedia Anda bayarkan.

 Warning

Jika Anda menentukan harga maksimum, instans Anda akan lebih sering diinterupsi daripada jika Anda memilih Tidak ada harga maksimum.

- Tidak ada harga maksimum: Instans Spot Anda akan diluncurkan pada harga Spot saat ini. Harga tidak akan pernah melebihi harga Sesuai Permintaan. (Direkomendasikan)
- Tetapkan harga maksimum Anda (per instans/jam): Anda dapat menentukan jumlah maksimum yang bersedia Anda bayarkan.
 - 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 [Riwayat harga Instans Spot](#).

- ii. Tipe permintaan: Permintaan Instans Spot yang dipilih menentukan apa yang terjadi jika Instans Spot Anda diinterupsi.

- Satu kali: Amazon EC2 menempatkan 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 fungsi hibernasi yang sama seperti Instans Sesuai Permintaan. 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 Amazon 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 [Pemecahan masalah peluncuran instans](#).

Old console

Untuk membuat permintaan Instans Spot menggunakan wizard peluncuran instans

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. Di halaman Pilih Amazon Machine Image (AMI), pilih AMI. Untuk informasi selengkapnya, lihat [Langkah 1: Pilih Amazon Machine Image \(AMI\)](#).
5. Di halaman Pilih Tipe Instans, pilih konfigurasi perangkat keras dan ukuran instans yang akan diluncurkan, lalu pilih Berikutnya: Konfigurasikan Detail Instans. Untuk informasi selengkapnya, lihat [Langkah 2: Pilih Tipe Instans](#).
6. Di halaman Konfigurasikan Detail Instans, konfigurasikan permintaan Instans Spot sebagai berikut:
 - Jumlah instans: Masukkan jumlah instans yang akan diluncurkan.

 Note

Amazon EC2 membuat permintaan terpisah untuk setiap Instans Spot.

- (Opsional) Untuk membantu memastikan bahwa Anda mempertahankan jumlah instans yang benar untuk menangani permintaan pada aplikasi, Anda dapat memilih Luncurkan ke Grup Auto Scaling untuk membuat konfigurasi 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](#).
- Opsi pembelian: Pilih Minta instans Spot untuk meluncurkan Instans Spot. Saat Anda memilih opsi ini, bidang berikut akan muncul.
- Harga Saat Ini: Harga Spot saat ini di setiap Zona Ketersediaan ditampilkan untuk tipe instans yang Anda pilih.
- (Opsional) Harga Maksimum: Anda dapat mengosongkan bidang tersebut atau menentukan jumlah maksimum yang bersedia Anda bayarkan.

 Warning

Jika Anda menentukan harga maksimum, instans Anda akan lebih sering diinterupsi daripada jika Anda memilih untuk mengosongkan bidang tersebut.

- Jika Anda menentukan harga maksimum yang kurang dari harga Spot, 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.
- Jika Anda membiarkan bidang kosong, Anda akan membayar harga Spot saat ini.
- Permintaan persisten: Pilih Permintaan persisten untuk mengirim ulang permintaan Instans Spot jika Instans Spot Anda diinterupsi.
- Perilaku Interupsi: Secara default, layanan Spot mengakhiri Instans Spot saat diinterupsi. Jika memilih Permintaan persisten, Anda dapat memilih layanan Spot untuk menghentikan

atau menghibernasi Instans Spot Anda saat diinterupsi. Untuk informasi selengkapnya, lihat [Perilaku interupsi Instance Spot](#).

- (Opsional) Permintaan berlaku hingga: Pilih Edit untuk menentukan kapan permintaan Instans Spot kedaluwarsa.

Untuk informasi selengkapnya tentang mengonfigurasi Instans Spot Anda, lihat [Langkah 3: Konfigurasi Detail Instans](#).

7. AMI yang Anda pilih mencakup satu atau lebih volume penyimpanan, termasuk volume perangkat root. Di halaman Add Storage, Anda dapat menentukan volume tambahan untuk dilampirkan ke instans dengan memilih Add New Volume. Untuk informasi selengkapnya, lihat [Langkah 4: Tambahkan Penyimpanan](#).
8. Pada halaman Tambahkan Tanda, tentukan [tanda](#) dengan memberikan kombinasi kunci dan nilai. Untuk informasi selengkapnya, lihat [Langkah 5: Tambahkan Tanda](#).
9. Di halaman Konfigurasi Grup Keamanan, gunakan grup keamanan untuk menentukan aturan firewall bagi instans Anda. Aturan ini menentukan lalu lintas jaringan yang masuk yang dikirim ke instans Anda. Semua lalu lintas lainnya diabaikan. (Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).) Pilih atau buat grup keamanan, lalu pilih Tinjau dan Luncurkan. Untuk informasi selengkapnya, lihat [Langkah 6: Konfigurasi Grup Keamanan](#).
10. Pada halaman Tinjau Peluncuran Instans, periksa detail dari instans, dan buat perubahan yang diperlukan dengan memilih tautan Edit yang sesuai. Saat Anda siap, pilih Luncurkan. Untuk informasi selengkapnya, lihat [Langkah 7: Tinjau Peluncuran Instans dan Pilih Pasangan Kunci](#).
11. Dalam kotak dialog Pilih pasangan kunci yang sudah ada atau buat pasangan kunci baru, Anda dapat memilih pasangan kunci yang sudah ada, atau membuat yang baru. Misalnya, pilih Pilih pasangan kunci yang ada, lalu pilih pasangan kunci yang Anda buat saat melakukan penyiapan. Untuk informasi selengkapnya, lihat [Pasangan kunci Amazon EC2 dan instans Amazon EC2](#).

 Important

Jika Anda memilih opsi Lanjutkan tanpa pasangan kunci, Anda tidak akan dapat terhubung ke instans, kecuali Anda memilih AMI yang dikonfigurasi agar pengguna dapat masuk dengan cara lain.

12. Untuk meluncurkan instans Anda, centang kotak penerimaan, lalu pilih Luncurkan Instans.

Jika instance gagal diluncurkan atau status langsung beralih ke `terminated` dari pada `running`, lihat [Pemecahan masalah peluncuran instans](#).

AWS CLI

Untuk membuat permintaan Instans Spot menggunakan [run-instances](#)

Gunakan perintah [run-instances](#) dan tentukan opsi Instans Spot di parameter `--instance-market-options`.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type t2.micro \  
  --count 5 \  
  --subnet-id subnet-08fc749671b2d077c \  
  --key-name MyKeyPair \  
  --security-group-ids sg-0b0384b66d7d692f9 \  
  --instance-market-options file://spot-options.json
```

Berikut adalah struktur data yang harus ditentukan dalam file JSON untuk `--instance-market-options`. 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 Instans Spot menggunakan [request-spot-instances](#)

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).

Temukan Instans Spot Anda

Amazon EC2 meluncurkan Instans Spot saat kapasitas tersedia. Instans Spot berjalan hingga diinterupsi atau Anda mengakhirinya sendiri.

Instans Spot muncul di halaman Instans di konsol, bersama dengan Instans Sesuai Permintaan. Gunakan prosedur berikut untuk menemukan Instans Spot Anda.

Console

Untuk menemukan Instans Spot menggunakan konsol

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 lifecycle=spot.
4. Untuk memverifikasi bahwa instance adalah Instans Spot, pilih instance, pilih tab Detail, dan periksa nilai Siklus Hidup. Nilai untuk Instans Spot adalah spot dan nilai untuk Instans On-Demand adalah normal.

AWS CLI

Untuk menemukan Instans Spot Anda menggunakan AWS CLI

Gunakan perintah [describe-instance](#) dengan opsi. `--filters`

```
aws ec2 describe-instances \  
  --filters "Name=instance-lifecycle,Values=spot"
```

Untuk menentukan apakah sebuah instance adalah Instans Spot

Gunakan perintah [describe-instance](#), menggunakan `--query` opsi untuk memeriksa nilai siklus hidup.

```
aws ec2 describe-instances \  
  --instance-ids i-0123a456700123456 \  
  --query "Reservations[*].Instances[*].InstanceLifecycle" \  
  --output text
```

Jika outputnya `spot`, instancenya adalah Instans Spot. Jika tidak ada output, instans adalah Instans On-Demand.

Gunakan prosedur berikut untuk menemukan Instans Spot yang diluncurkan dari permintaan Instance Spot atau Armada Spot tertentu.

Console

Untuk menemukan Instans Spot untuk permintaan menggunakan konsol

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 Armada Spot.

3. Jika permintaan Instans Spot terpenuhi, Kapasitas adalah ID dari 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 menggunakan AWS CLI

Gunakan [perintah describe-spot-instance-requests](#) dengan opsi. `--query`

```
aws ec2 describe-spot-instance-requests \  
  --query "SpotInstanceRequests[*].{ID:InstanceId}"
```

Berikut ini adalah output contoh:

```
[  
  {  
    "ID": "i-1234567890abcdef0"  
  },  
  {  
    "ID": "i-0598c7d356eba48d7"  
  }  
]
```

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`. Tindakan 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.

```
{
  "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 membuat, 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 menandai permintaan Instans Spot saat membuat, Amazon EC2 tidak mengevaluasi `spot-instances-request` sumber daya dalam pernyataan. `RunInstances`
- Jika Anda menandai permintaan Instans Spot saat membuat, Amazon EC2 akan mengevaluasi `spot-instances-request` sumber daya dalam pernyataan. `RunInstances`

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 membuat, Anda tidak perlu secara eksplisit mengizinkan `spot-instances-request` sumber daya; panggilan akan berhasil.
- Jika Anda menggunakan RunInstances untuk membuat permintaan Instans Spot dan bermaksud untuk menandai permintaan Instans Spot saat membuat, 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 bermaksud untuk menandai permintaan Instans Spot saat membuat, 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.

```
{
  "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

Console

Untuk menandai permintaan Instans Spot baru menggunakan konsol

1. Ikuti prosedur [Membuat permintaan Instans Spot](#).
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 [Membuat permintaan Instans Spot](#).

AWS CLI

Untuk menandai permintaan Instans Spot baru menggunakan AWS CLI

Untuk menandai permintaan Instans Spot saat Anda membuatnya, konfigurasi konfigurasi permintaan Instans Spot sebagai berikut:

- Tentukan tanda untuk permintaan Instans Spot menggunakan parameter `--tag-specification`.
- 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.

Dalam contoh berikut, permintaan Instans Spot ditandai dengan dua tanda: Kunci=Lingkungan dan Nilai=Produksi, serta Kunci=Pusat-Biaya dan Nilai=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}]'
```

Menandai permintaan Instans Spot yang ada

Console

Untuk menandai permintaan Instans Spot yang sudah ada menggunakan konsol

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 [Menambahkan dan menghapus tanda pada sumber daya individu](#).

AWS CLI

Untuk menandai permintaan Instans Spot atau Instance Spot yang ada menggunakan AWS CLI

Gunakan perintah [create-tags](#) untuk menandai sumber daya yang ada. Dalam contoh berikut, permintaan Instans Spot yang ada dan Instans Spot ditandai dengan Key = tujuan dan Value=pengujian.

```
aws ec2 create-tags \  
  --resources sir-08b93456 i-1234567890abcdef0 \  
  --tags Key=tujuan,Value=pengujian
```

```
--tags Key=purpose,Value=test
```

Melihat tanda permintaan Instans Spot

Console

Untuk melihat tanda 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 dan pilih tab Tanda.

AWS CLI

Untuk mendeskripsikan tag permintaan Instans Spot

Anda dapat melihat tag permintaan Instans Spot dengan menjelaskan 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-EXAMPLE1 \  
  --query "SpotInstanceRequests[*].Tags"
```

Berikut ini adalah output contoh.

```
[  
  [  
    {  
      "Key": "Environment",  
      "Value": "Production"  
    },  
    {  
      "Key": "Department",  
      "Value": "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 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.
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 Instans Spot menggunakan AWS CLI

Gunakan perintah [cancel-spot-instance-requests](#) untuk membatalkan permintaan Instans Spot yang ditentukan.

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

Menghentikan Instans Spot

Jika Anda tidak memerlukan Instans Spot sekarang, tetapi Anda ingin memulai ulang nanti tanpa kehilangan data yang tersimpan di volume Amazon EBS, Anda dapat menghentikannya. Langkah-langkah untuk menghentikan Instans Spot serupa dengan langkah-langkah untuk menghentikan Instans Sesuai Permintaan.

Note

Saat Instans Spot dihentikan, Anda dapat memodifikasi beberapa atribut instans, tetapi tidak untuk tipe 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 menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih Instans Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihat [the section called “Temukan Instans Spot Anda”](#).
4. Pilih Status instans, Hentikan instans.
5. Ketika diminta konfirmasi, pilih Berhenti.

AWS CLI

Untuk menghentikan Instance Spot menggunakan AWS CLI

Gunakan perintah [stop-instance untuk menghentikan Instans](#) Spot Anda secara manual.

```
aws ec2 stop-instances --instance-ids 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.
- Instans Spot adalah instans yang didukung EBS.
- 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 serupa dengan langkah-langkah untuk memulai Instans Sesuai Permintaan.

Console

Untuk memulai Instance Spot menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih Instans Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihat [the section called “Temukan Instans Spot Anda”](#).
4. Pilih Status instans, Mulai instans.

AWS CLI

Untuk memulai Instance Spot, AWS CLI

Gunakan perintah [start-instance untuk memulai Instans](#) Spot Anda secara manual.

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

Menghentikan Instans Spot

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 menggunakan konsol

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 Instans Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihat [the section called “Temukan Instans Spot Anda”](#).

5. Pilih Status instans, Akhiri instans.
6. Pilih Akhiri saat diminta untuk mengonfirmasi.

AWS CLI

Untuk menghentikan Instans Spot secara manual menggunakan AWS CLI

Gunakan perintah [terminate-instance untuk menghentikan Instans Spot](#) Anda secara manual.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7
```

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 [Membuat permintaan Instans Spot](#).

Important

Kami sangat tidak menyarankan menggunakan perintah [request-spot-instances](#) 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"
  }
}
```

Status permintaan 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.

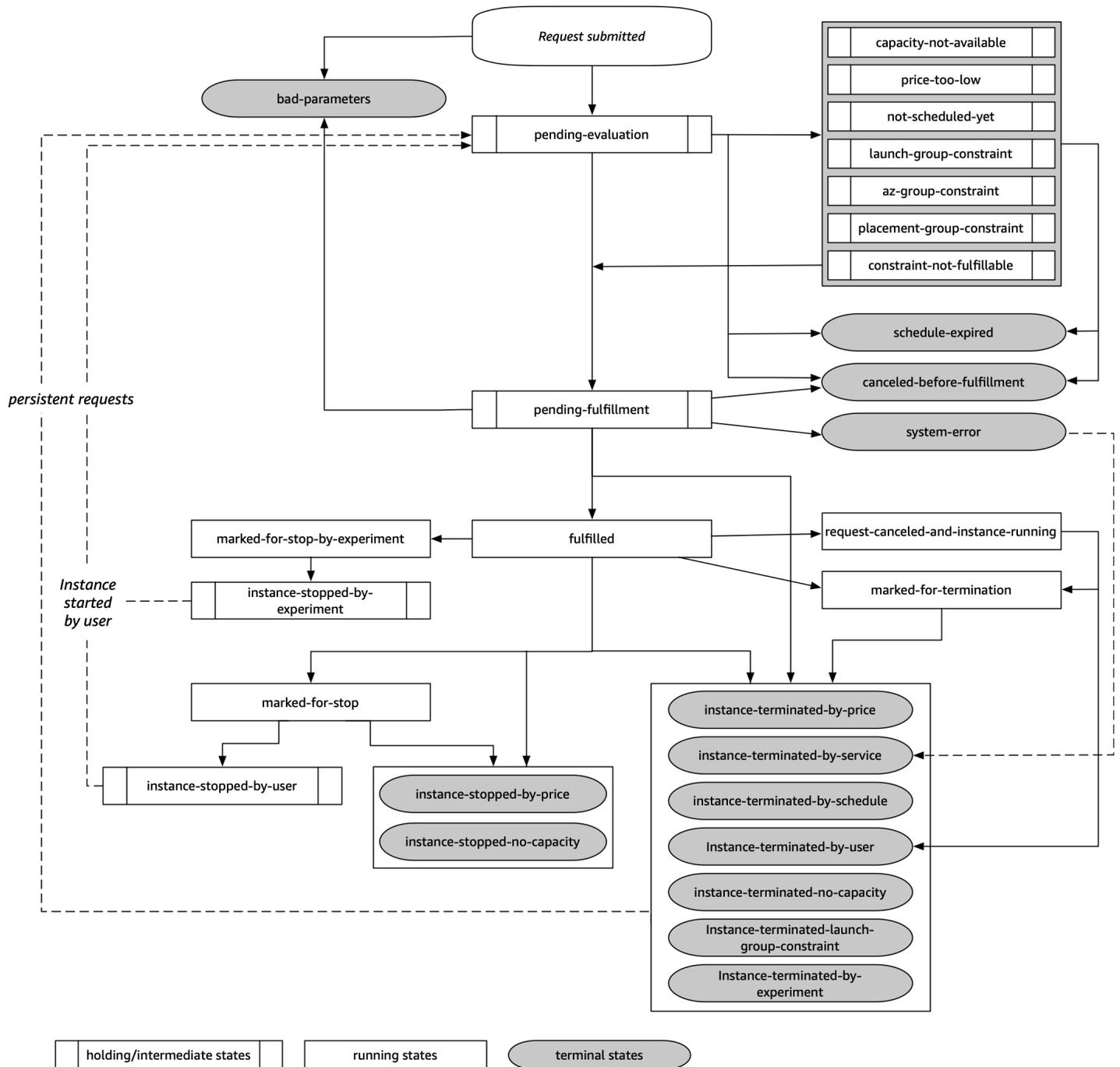
Pada setiap langkah proses—disebut juga dengan siklus hidup permintaan Spot—peristiwa spesifik menentukan status permintaan secara berurutan.

Daftar Isi

- [Siklus hidup permintaan Spot](#)
- [Dapatkan informasi status permintaan](#)
- [Kode status permintaan Spot](#)
- [Peristiwa Pemenuhan Permintaan Instans Spot EC2](#)

Siklus hidup 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

Kode status	Status permintaan	Status instans
<code>constraint-not-fulfillable</code>	open	Tidak berlaku

Evaluasi tertunda/terminal pemenuhan

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
<code>schedule-expired</code>	cancelled	Tidak berlaku
<code>cancel-before-fulfillment</code> ¹	cancelled	Tidak berlaku
<code>bad-parameters</code>	failed	Tidak berlaku
<code>system-error</code>	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
<code>pending-fulfillment</code>	<code>open</code>	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
<code>fulfilled</code>	<code>active</code>	<code>pending</code> → <code>running</code>
<code>fulfilled</code>	<code>active</code>	<code>stopped</code> → <code>running</code>

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
<code>marked-for-stop</code>	<code>active</code>	<code>stopping</code>
<code>instance-stopped-by-user</code> ¹	<code>disabled</code> atau <code>cancelled</code> ²	<code>stopped</code>

* 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.

Terminal terpenuhi

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
<code>request-canceled-and-instance-running</code>	<code>cancelled</code>	<code>running</code>
<code>marked-for-stop</code>	<code>active</code>	<code>running</code>
<code>marked-for-termination</code>	<code>active</code>	<code>running</code>
<code>instance-stopped-by-price</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-stopped-by-user</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-stopped-no-capacity</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-terminated-by-price</code>	<code>closed(satu kali),open (gigih)</code>	<code>terminated</code>
<code>instance-terminated-by-schedule</code>	<code>closed</code>	<code>terminated</code>
<code>instance-terminated-by-service</code>	<code>cancelled</code>	<code>terminated</code>

Kode status	Status permintaan	Status instans
<code>instance-terminated-by-user</code>	<code>closed</code> atau <code>cancelled</code> ¹	<code>terminated</code>
<code>instance-terminated-no-capacity</code>	<code>closed(satu kali)</code> , <code>open (gigih)</code>	<code>running</code> †
<code>instance-terminated-no-capacity</code>	<code>closed(satu kali)</code> , <code>open (gigih)</code>	<code>terminated</code>
<code>instance-terminate-d-launch-group-constraint</code>	<code>closed(satu kali)</code> , <code>open (gigih)</code>	<code>terminated</code>

* 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 menggunakan AWS Fault Injection Service untuk memulai interupsi Instans Spot sehingga Anda dapat menguji bagaimana aplikasi di Instans Spot 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.

Dapatkan informasi status permintaan

Anda bisa mendapatkan informasi status permintaan menggunakan AWS Management Console atau alat baris perintah.

Untuk mendapatkan informasi status permintaan menggunakan konsol

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.

Untuk mendapatkan informasi status permintaan menggunakan alat baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-spot-instance-requests](#) (AWS CLI)
- [Get-EC2SpotInstanceRequest](#) (AWS Tools for Windows PowerShell)

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, AML 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 tentang batas ini dan cara meminta peningkatan, lihat [Batas Amazon EBS](#) di Referensi Umum Amazon Web Services.

`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 yang berulang, silakan hubungi AWS Support untuk bantuan.

Peristiwa Pemenuhan Permintaan Instans Spot EC2

Ketika permintaan Instans Spot terpenuhi, Amazon EC2 mengirimkan peristiwa 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-1a2b3c4d",
  "instance-id": "i-1234567890abcdef0"
}
}
```

Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#).

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 penyeimbangan ulang tersedia sebagai EventBridge peristiwa dan sebagai item dalam [metadata instance pada 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.

Topik

- [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 interupsi Amazon EC2 Spot](#) di Panduan Pengguna Amazon EC2 Auto Scaling, serta [Penyeimbangan Ulang Kapasitas](#) untuk Armada EC2 dan [Penyeimbangan Ulang Kapasitas](#) untuk Armada Spot di 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 penyeimbangan ulang tersedia sebagai peristiwa yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon CloudWatch Events) dan sebagai metadata instans pada Instans Spot.

Pantau sinyal rekomendasi penyeimbangan kembali:

- [Gunakan Amazon EventBridge](#)
- [Gunakan metadata instans](#)

Gunakan Amazon EventBridge

Ketika sinyal rekomendasi penyeimbangan kembali 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 acara cocok dengan aturan.

Contoh berikut membuat EventBridge aturan untuk mengirim email, pesan teks, atau pemberitahuan push seluler setiap kali Amazon EC2 memancarkan sinyal rekomendasi penyeimbangan ulang. 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 penyeimbangan ulang

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 pesan application-to-person \(A2P\) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).
 - 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 peristiwa 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.

Gunakan perintah untuk sistem operasi Anda.

Linux

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/recommendations/rebalance
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

Windows

```
PS C:\> 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
- [Penyeimbangan Ulang Kapasitas](#) dalam topik Armada EC2 di panduan pengguna ini
- [Penyeimbangan Ulang Kapasitas](#) dalam topik Armada Spot di panduan pengguna ini

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 mengeklaim kembali Instans Spot, kami menyebut peristiwa ini sebagai Interupsi Instans Spot.

Saat menginterupsi Instans Spot, Amazon EC2 akan mengakhiri, menghentikan, atau melakukan hibernasi instans, tergantung pada apa yang Anda tentukan saat Anda membuat permintaan 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.

Instans Sesuai Permintaan yang ditentukan dalam Armada EC2 atau Armada Spot tidak dapat diinterupsi.

Daftar Isi

- [Alasan gangguan Instans Spot](#)
- [Perilaku interupsi Instance Spot](#)
- [Menghentikan Instans Spot yang terinterupsi](#)
- [Menghibernasi Instans Spot yang diinterupsi](#)
- [Mengakhiri Instans Spot yang diinterupsi](#)
- [Bersiaplah untuk interupsi Spot Instance](#)
- [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](#)

Alasan gangguan Instans Spot

Berikut ini adalah kemungkinan alasan Amazon EC2 menginterupsi Instans Spot Anda:

Kapasitas

Amazon EC2 dapat mengganggu menginterupsi Instans Spot Anda saat membutuhkannya kembali. EC2 mengklaim 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 menentukannya.

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.

Anda dapat melihat tingkat interupsi historis untuk tipe instans Anda di [Spot Instans Advisor](#).

Perilaku interupsi Instance Spot

Anda dapat menentukan bahwa Amazon EC2 harus melakukan salah satu dari hal berikut saat menginterupsi Instans Spot:

- [Menghentikan Instans Spot yang terinterupsi](#)
- [Menghibernasi Instans Spot yang diinterupsi](#)
- [Mengakhiri Instans Spot yang diinterupsi](#) (ini adalah perilaku default)

Menentukan perilaku interupsi

Anda dapat menentukan perilaku interupsi saat Anda membuat permintaan Spot. Jika Anda tidak menentukan perilaku interupsi, default-nya adalah Amazon EC2 mengakhiri Instans Spot saat diinterupsi.

Cara Anda menentukan perilaku interupsi berbeda-beda, bergantung pada cara Anda meminta Instans Spot.

- Jika Anda meminta Instans Spot menggunakan [wizard peluncuran instans](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Di wizard peluncuran instans, perluas Detail lanjutan dan pilih kotak centang Meminta Instans Spot. Pilih Sesuaikan. Dari Perilaku interupsi, pilih perilaku interupsi. Jika perilaku interupsi adalah hibernasi, Anda dapat memilih Aktifkan untuk Hentikan - Perilaku Hibernasi .
- Jika Anda meminta Instans Spot menggunakan CLI [run-instances](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Dalam konfigurasi permintaan, (`--instance-market-options`), untuk `InstanceInterruptionBehavior`, tentukan sebuah perilaku interupsi. Jika perilaku interupsi adalah hibernate, Anda dapat mengaktifkan hibernasi menggunakan parameter `--hibernation-options Configured=true`.
- Jika Anda mengonfigurasi Instans Spot di [Templat peluncuran](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Dalam templat peluncuran, perluas Detail lanjutan dan pilih kotak centang Meminta Instans Spot. Pilih Sesuaikan, lalu dari Perilaku interupsi, pilih perilaku interupsi.
- Jika Anda meminta Instans Spot menggunakan [Konsol Spot](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Pilih kotak centang Pertahankan kapasitas target, lalu dari Perilaku interupsi, pilih sebuah perilaku interupsi.
- Jika Anda mengonfigurasi Instans Spot dalam konfigurasi permintaan saat menggunakan CLI [create-fleet](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Untuk `InstanceInterruptionBehavior`, tentukan sebuah perilaku interupsi.
- Jika Anda mengonfigurasi Instans Spot dalam konfigurasi permintaan saat menggunakan CLI [request-spot-fleet](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Untuk `InstanceInterruptionBehavior`, tentukan sebuah perilaku interupsi.
- Jika Anda mengonfigurasi Instans Spot menggunakan CLI [request-spot-instances](#), Anda dapat menentukan perilaku interupsi sebagai berikut: Untuk `--instance-interruption-behavior`, tentukan sebuah perilaku interupsi.

Note

Kami sangat tidak menyarankan penggunaan perintah [request-spot-fleet](#) dan [request-spot-instances](#) untuk meminta Instans Spot karena perintah itu adalah API warisan tanpa investasi yang direncanakan. Untuk informasi selengkapnya, lihat [Metode permintaan Spot mana yang terbaik untuk digunakan?](#)

Menghentikan Instans Spot yang terinterupsi

Anda dapat menentukan bahwa Amazon EC2 akan menghentikan Instans Spot Anda ketika instans diinterupsi. Untuk informasi selengkapnya, lihat [Menentukan perilaku interupsi](#).

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).
- Untuk Instans Spot yang diluncurkan oleh Armada EC2 atau Armada Spot bertipe *maintain*: Setelah Instans Spot terputus, Amazon EC2 meluncurkan instans pengganti untuk mempertahankan kapasitas target. Amazon EC2 menemukan kolam kapasitas Spot terbaik berdasarkan strategi alokasi yang ditentukan (`lowestPrice`, `diversified`, atau `InstancePoolsToUseCount`); layanan tersebut tidak memprioritaskan kolam dengan instans yang dihentikan sebelumnya. Nantinya, jika strategi alokasi mengarah ke kolam yang berisi instans yang dihentikan sebelumnya, Amazon EC2 memulai ulang instans yang dihentikan untuk memenuhi kapasitas target.

Misalnya, pertimbangkan Armada Spot dengan strategi alokasi `lowestPrice`. Pada peluncuran awal, kolam `c3.large` memenuhi kriteria `lowestPrice` untuk spesifikasi peluncuran. Nantinya, jika instans `c3.large` diinterupsi, Amazon EC2 akan menghentikan instans dan mengisi kembali kapasitas dari kolam lain yang sesuai dengan strategi `lowestPrice`. Kali ini, kolam tersebut kebetulan berupa kolam `c4.large` dan Amazon EC2 meluncurkan instans `c4.large` untuk memenuhi kapasitas target. Demikian pula, Armada Spot bisa pindah ke kolam `c5.large` di waktu berikutnya. Di setiap transisi ini, Amazon EC2 tidak memprioritaskan kolam dengan instans yang dihentikan sebelumnya, melainkan hanya memprioritaskan strategi alokasi yang ditentukan. Strategi `lowestPrice` dapat mengarah kembali ke kolam dengan instans yang

dihentikan sebelumnya. Misalnya, jika instans diinterupsi di kolam `c5.large` dan strategi `lowestPrice` mengarahkannya kembali ke kolam `c3.large` atau `c4.large`, instans yang dihentikan sebelumnya akan dimulai ulang untuk memenuhi kapasitas target.

- 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 boot ulang, penghentian, hibernasi, dan pengakhiran](#).

Prasyarat

Untuk menghentikan Instans Spot yang diinterupsi, prasyarat berikut harus tersedia:

Tipe permintaan spot

Tipe permintaan Instans Spot – Harus `persistent`. Anda tidak dapat menentukan grup peluncuran dalam permintaan Instans Spot.

Tipe permintaan Armada EC2 atau Armada Spot – Harus `maintain`.

Tipe volume root

Harus berupa volume EBS, bukan volume penyimpanan instans.

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 kini menawarkan pengalaman hibernasi yang sama untuk Instans Spot seperti yang saat ini tersedia untuk Instans Sesuai Permintaan. 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](#)
- [Hibernasi yang diprakarsai pengguna](#)

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 [Menentukan perilaku interupsi](#).

Bersiaplah untuk interupsi Spot Instance

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 Instans Sesuai Permintaan, kemudian mengakhiri sendiri instans sesuai permintaan itu.
- Jalankan eksperimen injeksi kesalahan terkontrol 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 Instans 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 interupsi Instans Spot didukung di semua Wilayah AWS kecuali Asia Pasifik (Jakarta), Asia Pasifik (Osaka), China (Beijing), China (Ningxia), dan Timur Tengah (UEA).

Topik

- [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 lainnya untuk menguji interupsi Instans Spot, Anda dapat membuat eksperimen sendiri menggunakan konsol. AWS FIS

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

Jumlahnya tidak dapat melebihi jumlah Instans Spot di armada atau [kuota](#) Anda untuk jumlah Instans Spot yang AWS FIS dapat diinterupsi 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 lanjutan lainnya untuk menguji interupsi Instans Spot menggunakan konsol AWS FIS

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.

AWS FIS Konsol 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 [rekomen-dasi penyeimbangan kembali instans](#).
- [Pemberitahuan interupsi Instans Spot](#) dikeluarkan dua menit sebelum AWS FIS menginterupsi instans Anda.

- Setelah dua menit, Instans Spot akan diinterupsi.
- Instance Spot yang dihentikan oleh AWS FIS tetap berhenti 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-interruptions	Setiap Wilayah yang didukung: 5	Ya	Jumlah maksimum Instans Spot yang aws:ec2: send-spot-instance-interruptions dapat menargetkan saat Anda mengidentifikasi

Nama	Default	Dapat disesuaikan	Deskripsi
			target menggunakan tag, per percobaan.

Anda dapat meminta penambahan kuota. Untuk informasi selengkapnya, lihat [Meminta peningkatan kuota](#) di Panduan Pengguna Kuota Layanan.

Untuk melihat semua kuota AWS FIS, buka konsol [Service Quotas](#). 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 [metadana instance pada Instans](#) Spot. Pemberitahuan interupsi dipancarkan dengan upaya yang terbaik.

EC2 Spot Instance interruption notice

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). Acara ini dapat dideteksi oleh Amazon EventBridge. Untuk informasi selengkapnya tentang EventBridge peristiwa, 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 Instans Spot adalah. `arn:aws:ec2:availability-zone:instance/instance-id` Format ini berbeda dari format [ARN sumber daya EC2](#).

instance-action

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 Metadata Service Version 2 (IMDSv2) sebagai berikut. `instance-action`

Gunakan perintah untuk sistem operasi Anda.

Linux

```
[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/spot/instance-action`
```

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/meta-data/spot/instance-action
```

Item `instance-action` menentukan tindakan dan perkiraan waktu, dalam UTC, kapan tindakan akan terjadi.

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.

`termination-time`

Item ini dipertahankan untuk kompatibilitas mundur; Anda seharusnya menggunakan `instance-action`.

[Jika Instans Spot ditandai untuk dihentikan oleh Amazon EC2 \(baik karena gangguan Instans Spot di mana perilaku interupsi disetel `terminate`, atau karena pembatalan permintaan Instans Spot persisten\), `termination-time` item tersebut ada dalam metadata instans Anda.](#) Jika tidak, item itu tidak ada. Anda dapat mengambil `termination-time` menggunakan IMDSv2 sebagai berikut.

Gunakan perintah untuk sistem operasi Anda.

Linux

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
[ec2-user ~]$ 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

```
PS C:\> [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
```

`termination-time` menentukan perkiraan waktu dalam UTC kapan instance akan menerima sinyal shutdown. Berikut ini adalah output contoh.

```
2015-01-05T18:02:00Z
```

Jika Amazon EC2 tidak bersiap untuk menghentikan instance (baik karena tidak ada interupsi 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

Di konsol, panel Instans menampilkan semua instans, termasuk Instans Spot. Siklus hidup instans dari instans Spot adalah `spot`. Status instans dari Instans Spot bisa berupa `stopped` atau `terminated`, tergantung pada perilaku interupsi yang Anda konfigurasi. Untuk instans Spot hibernasi, status instans adalah `stopped`.

Untuk menemukan Instans Spot yang diinterupsi menggunakan konsol

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 konfigurasi.
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`
6. Untuk detail tambahan tentang alasan interupsi, periksa kode status permintaan Spot. Untuk informasi selengkapnya, lihat [the section called "Status permintaan spot"](#).

Untuk menemukan Instans Spot yang terputus menggunakan AWS CLI

Anda dapat membuat daftar Instans Spot yang diinterupsi menggunakan perintah [describe-instances](#) dengan parameter `--filters`. Untuk mendaftar hanya ID instans di output, tambahkan parameter `--query`.

Jika perilaku interupsi instans adalah untuk mengakhiri Instans Spot, gunakan perintah 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 instans adalah menghentikan Instans Spot, gunakan perintah 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"
```

Menentukan apakah Amazon EC2 mengakhiri Instans Spot

Jika Instans Spot dihentikan, Anda dapat menggunakannya CloudTrail untuk melihat apakah Amazon EC2 menghentikan Instans Spot. Di AWS CloudTrail, nama peristiwa `BidEvictedEvent` menunjukkan bahwa Amazon EC2 mengakhiri Instans Spot.

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. Di drop-down filter, pilih Nama acara, dan kemudian di bidang filter di sebelah kanan, masukkan `BidEvictedAcara`.
4. Pilih `BidEvictedAcara` dalam daftar yang dihasilkan untuk melihat detailnya. Pada Catatan peristiwa, Anda dapat menemukan ID instans.

Untuk informasi selengkapnya tentang penggunaan CloudTrail, lihat [Log panggilan API Amazon EC2 menggunakan AWS CloudTrail](#).

Penagihan untuk Instans Spot yang diinterupsi

Ketika Instans Spot terganggu, Anda dikenai biaya untuk instans dan penggunaan volume EBS. Selain itu, Anda mungkin akan dikenai biaya lainnya 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
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.

Biaya lainnya

Jika Instans Spot Anda yang sedang berjalan dikenakan biaya untuk layanan lain, seperti untuk transfer data, alamat IP Elastis, atau penggunaan layanan AWS terkelola lainnya, Anda akan ditagih untuk penggunaannya. Ini terlepas dari siapa yang mengganggu Instans Spot atau kapan Instans Spot terganggu. Meskipun Anda tidak dikenai biaya untuk penggunaan Instans Spot saat Amazon EC2 mengganggu Instans Spot Anda dalam satu jam pertama, Anda dapat dikenai biaya lainnya.

Untuk informasi selengkapnya tentang biaya lainnya, lihat [Harga Sesuai Permintaan Amazon EC2](#).

Skor penempatan Spot

Fitur skor penempatan Spot dapat merekomendasikan AWS Wilayah atau Zona Ketersediaan 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.

Keuntungan

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 menyimulasikan kebutuhan kapasitas Spot di masa mendatang sehingga Anda dapat memilih Wilayah yang optimal untuk perluasan beban kerja berbasis Spot Anda.
- Untuk menemukan kombinasi tipe instans yang optimal untuk memenuhi kebutuhan kapasitas Spot Anda.

Topik

- [Biaya](#)
- [Cara kerja skor penempatan Spot](#)
- [Batasan](#)
- [Izin IAM yang diperlukan](#)
- [Hitung skor penempatan Spot](#)
- [Contoh konfigurasi](#)

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.

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 `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.

Langkah 3: Tinjau rekomendasi

Skor penempatan Spot untuk setiap Wilayah atau Zona Ketersediaan dihitung berdasarkan kapasitas target, komposisi tipe instans, tren penggunaan Spot historis dan saat ini, serta 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.

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 Auto Scaling dengan serangkaian kebutuhan pada tipe instans yang digunakan](#), [Pemilihan tipe instans berbasis atribut untuk Armada EC2](#), dan [Pemilihan tipe instans berbasis atribut untuk Armada Spot](#).

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.

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.

Izin IAM yang diperlukan

Secara default, identitas IAM (pengguna, peran, atau grup) tidak memiliki izin untuk menggunakan fitur Skor penempatan Spot. Untuk mengizinkan identitas IAM menggunakan fitur skor penempatan Spot, Anda harus membuat kebijakan IAM yang memberikan izin untuk menggunakan tindakan EC2 API `ec2:GetSpotPlacementScores`. 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`.

```
{
  "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, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.

- (Tidak disarankan) Pasang 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 dengan menggunakan konsol Amazon EC2 atau AWS CLI.

Topik

- [Hitung skor penempatan Spot dengan menentukan atribut instans \(konsol\)](#)
- [Hitung skor penempatan Spot dengan menentukan tipe instans \(konsol\)](#)
- [Hitung skor penempatan Spot \(AWS CLI\)](#)

Hitung skor penempatan Spot dengan menentukan atribut instans (konsol)

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 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 penjelasan tentang setiap atribut dan nilai default-nya, lihat [get-spot-placement-scores](#) dalam Referensi Baris Perintah Amazon EC2.

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) Untuk menampilkan skor penempatan Spot untuk Zona Ketersediaan di Wilayah Region yang ditampilkan, 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.
15. (Opsional) Untuk mengedit kebutuhan komputasi Anda dan mendapatkan skor penempatan baru, pilih Edit, buat penyesuaian yang diperlukan, lalu pilih Hitung skor penempatan.

Hitung skor penempatan Spot dengan menentukan tipe instans (konsol)

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 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) Untuk menampilkan skor penempatan Spot untuk Zona Ketersediaan di Wilayah Region yang ditampilkan, 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.
11. (Opsional) Untuk mengedit daftar tipe instans dan mendapatkan skor penempatan baru, pilih Edit, buat penyesuaian yang diperlukan, lalu pilih Hitung skor penempatan.

Hitung skor penempatan Spot (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
```

Output yang diharapkan

```
{  
  "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": ""
}

```

2. Buat file konfigurasi JSON menggunakan output dari langkah sebelumnya, dan konfigurasi 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 `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.

- e. 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 penjelasan tentang setiap atribut dan nilai default-nya, lihat [get-spot-placement-scores](#) dalam Referensi Baris Perintah Amazon EC2.

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 output jika `SingleAvailabilityZone` diatur ke `false` atau dihilangkan (jika dihilangkan, defaultnya adalah `false`) - daftar Wilayah dengan skornya akan ditampilkan

```
"SpotPlacementScores": [  
  {  
    "Region": "us-east-1",  
    "Score": 7
```

```

    },
    {
      "Region": "us-west-1",
      "Score": 5
    },
    ...

```

Contoh output jika `SingleAvailabilityZone` diatur ke `true` — daftar Zona Ketersediaan dengan skornya akan ditampilkan

```

"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 mengetahui daftar atribut, lihat [get-spot-placement-scores](#) dalam Referensi Baris Perintah Amazon EC2.

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
    }
  }
}
```

Umpan data Spot Instans

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 sekali dalam satu jam, dan setiap jam penggunaan 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 feed data Instance Spot per Akun AWS. Jika Anda tidak menjalankan Instans Spot selama jam tertentu, Anda tidak menerima file data feed untuk jam itu.

Umpan data Instans Spot didukung di semua AWS Wilayah kecuali China (Beijing), China (Ningxia), AWS GovCloud (AS), dan [Wilayah yang dinonaktifkan secara default](#).

Daftar Isi

- [Nama dan format file umpan data](#)
- [Persyaratan bucket Amazon S3](#)

- [Berlangganan ke umpan data Instans Spot Anda](#)
- [Jelaskan 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 **my-bucket-name** dan prefiks Anda adalah **my-prefix**, nama file Anda mirip dengan yang berikut ini:

```
my-bucket-name.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 instans 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 contoh, SpotUsage : c1.medium
Operation	Produk yang ditagihkan. Untuk Instans Spot Linux, bidang ini diatur ke RunInstances . Untuk Instans Spot Windows, bidang ini diatur ke

Bidang	Deskripsi
	RunInstances:0002 . Penggunaan spot dikelompokkan menurut Zona Ketersediaan.
InstanceID	ID Instans Spot yang menghasilkan penggunaan instans ini.
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 ember harus memberikan izin Akun AWS ini kepada Anda.
- Saat Anda berlangganan umpan data, izin ini digunakan untuk memperbarui bucket ACL untuk memberikan izin akun FULL_CONTROL umpan AWS data. Akun umpan AWS 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 Pengguna Amazon CloudWatch Logs.

Note

Jika Anda memperbarui ACL dan menghapus izin untuk akun umpan AWS 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 umpan AWS data memiliki izin baca dan tulis.
- Jika Anda menerapkan ACL yang dinonaktifkan untuk bucket Anda, tambahkan kebijakan bucket yang memungkinkan pengguna dengan kontrol penuh untuk menulis ke bucket. Untuk informasi selengkapnya, lihat [Meninjau dan memperbarui kebijakan bucket](#).
- Jika Anda menghapus langganan umpan data, Amazon EC2 tidak menghapus izin baca dan tulis untuk akun umpan AWS data di bucket atau file data. Anda harus menghapus izin ini sendiri.
- Anda harus menggunakan kunci yang dikelola pelanggan jika mengenkripsi bucket Amazon S3 menggunakan enkripsi sisi server dengan kunci yang disimpan di (AWS KMS SSE-KMS). AWS Key Management Service Untuk informasi selengkapnya, lihat [enkripsi sisi server bucket Amazon S3 di Panduan Pengguna Amazon Logs. CloudWatch](#)

Note

Untuk umpan data Instans Spot, sumber daya yang menghasilkan file S3 bukan lagi Amazon CloudWatch Logs. Oleh karena itu, Anda harus menghapus bagian `aws:SourceArn` dari kebijakan izin bucket S3 dan dari kebijakan KMS.

Berlangganan ke umpan data Instans Spot Anda

Untuk berlangganan umpan data Anda, gunakan perintah [create-spot-datafeed-subscription](#).

```
aws ec2 create-spot-datafeed-subscription \  
  --bucket my-bucket-name \  
  [--prefix my-prefix]
```

Contoh Output

```
{
```

```
"SpotDatafeedSubscription": {
  "OwnerId": "111122223333",
  "Bucket": "my-bucket-name",
  "Prefix": "my-prefix",
  "State": "Active"
}
```

Jelaskan umpan data Instans Spot Anda

Untuk menjelaskan umpan data Anda, gunakan perintah [describe-spot-datafeed-subscription](#).

```
aws ec2 describe-spot-datafeed-subscription
```

Contoh Output

```
{
  "SpotDatafeedSubscription": {
    "OwnerId": "123456789012",
    "Prefix": "spotdata",
    "Bucket": "my-s3-bucket",
    "State": "Active"
  }
}
```

Melihat data di umpan data Anda

Di AWS Management Console, terbuka AWS CloudShell. Gunakan perintah [s3 sync](#) berikut guna mendapatkan file .gz dari bucket S3 untuk umpan data Anda dan simpan di folder yang Anda tentukan.

```
aws s3 sync s3://my-s3-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

Untuk menghapus umpan data Anda, gunakan perintah [delete-spot-datafeed-subscription](#).

```
aws ec2 delete-spot-datafeed-subscription
```

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 tertunda terpenuhi, permintaan tidak lagi dihitung terhadap kuota karena instans yang sedang berjalan dihitung terhadap 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 tipe kuota berikut untuk Instans Spot:

- Semua Permintaan Instans Spot DL
- Semua Permintaan Instans Spot F
- Semua Permintaan Instans Spot G dan VT
- Semua Permintaan Instans Spot Inf
- Semua Permintaan Instans Spot P
- Semua Permintaan Instans Spot Standar (A, C, D, H, I, M, R, T, Z)

- Semua Permintaan Instans Spot Trn
- Semua Permintaan Instans Spot X

Setiap jenis kuota menentukan jumlah maksimum vCPU untuk satu atau beberapa keluarga instans. Untuk informasi tentang berbagai keluarga, generasi, dan ukuran instans, lihat [Tipe Instans 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).

Tugas

- [Pantau kuota dan penggunaan Instans Spot](#)
- [Meminta peningkatan kuota](#)

Pantau kuota dan penggunaan Instans Spot

Anda dapat melihat dan mengelola kuota Instans Spot menggunakan yang berikut ini:

- [Halaman Kuota Layanan](#) Amazon EC2 di konsol Kuota Layanan
- Kuota [get-service-](#) AWS CLI

Untuk informasi selengkapnya, lihat [Kuota layanan Amazon EC2](#) dan [Melihat kuota layanan](#) di Panduan Pengguna Service Quotas.

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 [Service Quotas dan CloudWatch alarm Amazon](#) di Panduan Pengguna Service Quotas di Panduan Pengguna Amazon. CloudWatch

Meminta peningkatan kuota

Meskipun Amazon EC2 secara otomatis meningkatkan kuota Instans Spot berdasarkan penggunaan Anda, Anda dapat meminta kenaikan 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](#).

Instance performa yang dapat melonjak

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 mengakumulasi 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 mengakumulasi 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.

Host Khusus

Host Khusus Amazon EC2 adalah server fisik yang sepenuhnya didedikasikan untuk Anda gunakan. Anda dapat memilih untuk berbagi kapasitas instans dengan AWS akun lain. Untuk informasi selengkapnya, lihat [Bekerja dengan Host Khusus bersama](#).

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 [Pahami penempatan otomatis dan afinitas](#).

Tuan Rumah 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, socket, atau inti fisik, tunduk pada persyaratan lisensi Anda.

Jika Anda memerlukan instans Anda untuk berjalan pada perangkat keras khusus, tetapi Anda tidak memerlukan visibilitas atau kontrol atas penempatan instans, dan Anda tidak perlu menggunakan lisensi perangkat lunak per-socket 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 instans yang sepenuhnya didedikasikan untuk Anda gunakan.	Server fisik yang didedikasikan untuk satu akun pelanggan.
Pembagian kapasitas instans	Dapat berbagi kapasitas instans dengan akun lain.	Tidak didukung
Penagihan	Tagihan per host	Tagihan per instans
Visibilitas socket, inti, dan ID host	Memberikan visibilitas dalam jumlah socket 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

	Host Khusus	Instans Khusus
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 .	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](#).

Daftar Isi

- [Konfigurasi kapasitas instans](#)
- [Bawa lisensi Anda sendiri](#)
- [Harga dan penagihan](#)
- [Instans T3 yang dapat melonjak pada Host Khusus](#)
- [Larangan Host Khusus](#)
- [Bekerja dengan Host Khusus](#)
- [Bekerja dengan Host Khusus bersama](#)
- [Tuan Rumah Khusus di AWS Outposts](#)
- [Pemulihan host](#)
- [Pemeliharaan host](#)
- [Lacak perubahan konfigurasi](#)

Konfigurasi kapasitas instans

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 ($48 \text{ core fisik} / 8 \text{ core per instans} = 6 \text{ instans}$).

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 tipe instans R5 apa pun, seperti `r5.large`, `r5.xlarge`, `r5.2xlarge`, dan `r5.4xlarge`, pada host tersebut, hingga kapasitas inti fisik host.

Keluarga instans berikut mendukung Host Khusus dengan dukungan beberapa tipe instans:

- Tujuan umum: A1, M5, M5n, M6i, dan T3
- Komputasi dioptimalkan: C5, C5n, dan C6i
- Memori yang dioptimalkan: R5, R5n, dan R6i

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 instans) dan tiga `r5.4xlarge` instans (8 core x 3 instans), instans tersebut mengkonsumsi total 32 core, dan Anda dapat menjalankan kombinasi R5 instans selama tidak melebihi 16 core yang tersisa.

Namun, untuk setiap keluarga instans, ada batas pada jumlah instans yang dapat dijalankan untuk setiap ukuran instans. Misalnya, Host R5 Khusus mendukung maksimal 2 `r5.8xlarge` instans, yang menggunakan 32 inti fisik. Dalam hal ini, R5 instans tambahan dengan ukuran yang lebih kecil kemudian dapat digunakan untuk mengisi host untuk kapasitas intinya. Untuk jumlah ukuran instans yang didukung untuk setiap keluarga instans, lihat [Tabel Konfigurasi Host Khusus](#).

Tabel berikut menunjukkan contoh kombinasi tipe instans:

Keluarga instans	Contoh kombinasi ukuran instans
R5	<ul style="list-style-type: none"> • Contoh 1: 4 x <code>r5.4xlarge</code> + 4 x <code>r5.2xlarge</code> • Contoh 2: 1 x <code>r5.12xlarge</code> + 1 x <code>r5.4xlarge</code> + 1 x <code>r5.2xlarge</code> + 5 x <code>r5.xlarge</code> + 2 x <code>r5.large</code>
C5	<ul style="list-style-type: none"> • Contoh 1: 1 x <code>c5.9xlarge</code> + 2 x <code>c5.4xlarge</code> + 1 x <code>c5.xlarge</code> • Contoh 2: 4 x <code>c5.4xlarge</code> + 1 x <code>c5.xlarge</code> + 2 x <code>c5.large</code>
M5	

Keluarga instans	Contoh kombinasi ukuran instans	
	<ul style="list-style-type: none">• Contoh 1: 4 x m5.4xlarge + 4 x m5.2xlarge• Contoh 2: 1 x m5.12xlarge + 1 x m5.4xlarge + 1 x m5.2xlarge + 5 x m5.xlarge + 2 x m5.large	

Pertimbangan

Ingatlah hal-hal berikut ini saat bekerja dengan Host Khusus yang mendukung banyak tipe instans:

- Dengan Host Khusus tipe-N, seperti C5n, M5n, dan R5n, Anda tidak dapat mencampur ukuran instans yang lebih kecil (2xlarge dan yang lebih kecil) dengan ukuran instans yang lebih besar (4xlarge dan yang lebih besar, termasuk meta1). Jika Anda memerlukan ukuran instans yang lebih kecil dan lebih besar pada Host Khusus tipe-N secara bersamaan, Anda harus mengalokasikan host terpisah untuk ukuran instans yang lebih kecil dan lebih besar.
- 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.

Bawa lisensi Anda sendiri

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 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 gambar mesin Anda dapat digunakan dalam Amazon EC2, impor gambar menggunakan VM Import/Export. Untuk informasi tentang cara mengimpor gambar mesin Anda, lihat [Panduan Pengguna VM Import/Export](#).

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 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](#).

Harga dan penagihan

Harga untuk Host Khusus bervariasi menurut opsi pembayaran.

Opsi pembayaran

- [Host Khusus Sesuai Permintaan](#)
- [Reservasi Host Khusus](#)
- [Savings Plans](#)
- [Harga untuk Windows Server pada Host Khusus](#)

Host Khusus Sesuai Permintaan

Penagihan Sesuai Permintaan secara otomatis diaktifkan saat Anda mengalokasikan Host Khusus ke akun Anda.

Harga Sesuai Permintaan 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 harga Sesuai Permintaan, lihat [Harga Sesuai Permintaan Host Khusus Amazon EC2](#).

Anda dapat melepas Host Khusus Sesuai Permintaan kapan saja untuk berhenti mengakumulasi biayanya. Untuk informasi tentang pelepasan Host Khusus, lihat [Melepas Host Khusus](#).

Reservasi Host Khusus

Reservasi Host Khusus memberikan diskon penagihan dibandingkan dengan menjalankan Host Khusus Sesuai Permintaan. 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](#).

Savings Plans

Savings Plans adalah model penetapan harga fleksibel yang menawarkan penghematan signifikan atas Instans Sesuai Permintaan. Dengan Savings Plans, Anda mermbuat 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](#).

Instans T3 yang dapat melonjak pada Host Khusus

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 mengakumulasi 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 diakumulasi 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 memanfaatkan sumber daya host secara efisien dan memberikan kinerja instans terbaik, algoritme penempatan instans Amazon EC2 secara otomatis menghitung jumlah instans dan kombinasi ukuran instans yang didukung 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.

Jenis instans	vCPU	Memori (GiB)	Pemanfaatan CPU acuan per vCPU	Bandwidth lonjakan jaringan (Gbps)	Bandwidth lonjakan Amazon EBS (Mbps)	Jumlah maksimum instans per Host Khusus
t3.nano	2	0,5	5%	5	Hingga 2.085	192

Jenis instan	vCPU	Memori (GiB)	Pemanfaatan CPU acuan per vCPU	Bandwidth lonjakan jaringan (Gbps)	Bandwidth lonjakan Amazon EBS (Mbps)	Jumlah maksimum instans per Host Khusus
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](#).

Larangan Host Khusus

Sebelum Anda mengalokasikan Host Khusus, perhatikan batasan dan larangan berikut:

- Untuk menjalankan RHEL, SUSE Linux, dan SQL Server di Host Khusus, Anda harus membawa AMI sendiri. RHEL, SUSE Linux, dan SQL Server AMI yang ditawarkan oleh AWS atau yang tersedia tidak AWS Marketplace dapat digunakan dengan Host Khusus. Untuk informasi lebih lanjut tentang cara membuat AMI Anda sendiri, lihat [Bawa lisensi Anda sendiri](#).

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`). RHEL dan SUSE Linux AMI 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](#).

- Instans yang berjalan di Host Khusus hanya dapat diluncurkan di VPC.
- Grup Auto Scaling didukung saat menggunakan templat peluncuran yang menentukan grup sumber daya host. Untuk informasi selengkapnya, lihat [Membuat template peluncuran menggunakan setelan lanjutan](#) di Panduan Pengguna Auto Scaling 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 dalam grup penempatan.
- Jika Anda mengalokasikan host untuk tipe instans tervirtualisasi, Anda tidak dapat mengubah tipe instans menjadi tipe instans `.metal` 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 `m5.metal` instans, Anda tidak dapat mengubah tipe instans menjadi `m5.large`.

Bekerja dengan Host Khusus

Untuk menggunakan Host Khusus, pertama alokasikan host untuk digunakan di akun Anda. Anda kemudian meluncurkan instans ke host dengan menentukan penghunian host untuk instans tersebut. Anda harus memilih host tertentu untuk peluncuran instans, atau Anda dapat mengizinkan instans untuk diluncurkan ke host mana pun yang mengaktifkan penempatan otomatis dan cocok dengan tipe instansnya. Saat sebuah instans dihentikan dan dimulai ulang, pengaturan Afinitas host menentukan apakah instans akan dimulai ulang pada host yang sama atau berbeda.

Jika Anda tidak lagi membutuhkan host Sesuai Permintaan, Anda dapat menghentikan instans yang berjalan pada host tersebut, mengarahkannya untuk diluncurkan pada host yang berbeda, lalu melepaskan host tersebut.

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 .

Daftar Isi

- [Alokasikan Host Khusus](#)
- [Luncurkan instans pada Host Khusus](#)
- [Luncurkan sebuah instans ke dalam grup sumber daya host](#)
- [Pahami penempatan otomatis dan afinitas](#)
- [Memodifikasi penempatan otomatis Host Khusus](#)
- [Mengubah tipe instans yang didukung](#)
- [Memodifikasi penghunian dan afinitas instans](#)
- [Melihat Host Khusus](#)
- [Tandai Host Khusus](#)
- [Memantau Host Khusus](#)
- [Melepas Host Khusus](#)
- [Membeli Reservasi Host Khusus](#)
- [Melihat reservasi Host Khusus](#)
- [Menandai Reservasi Host Khusus](#)

Alokasikan Host Khusus

Untuk mulai menggunakan Host Khusus, Anda harus mengalokasikan Host Khusus di akun Anda menggunakan konsol Amazon EC2 atau alat baris perintah. 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](#).

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 [Pahami penempatan otomatis dan afinitas](#).
7. Untuk mengaktifkan pemulihan host untuk Host Khusus, pada Pemulihan host, pilih Aktifkan. Untuk informasi selengkapnya, lihat [Pemulihan host](#).
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](#) AWS CLI . Perintah berikut mengalokasikan Host Khusus yang mendukung banyak tipe instans dari keluarga instans m5 di Zona Ketersediaan us-east-1a. Host juga mengaktifkan 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
```

Perintah berikut mengalokasikan Host Khusus yang mendukung instans tidak bertargetm4.large diluncurkan di Zona Ketersediaan eu-west-1a, mengaktifkan pemulihan host, dan menerapkan tanda dengan kunci purpose dan nilai production.

```
aws ec2 allocate-hosts --instance-type "m4.large" --availability-zone "eu-west-1a"
--auto-placement "on" --host-recovery "on" --quantity 1 --tag-specifications
'ResourceType=dedicated-host,Tags=[{Key=purpose,Value=production}]'
```

PowerShell

Untuk mengalokasikan Host Khusus

Gunakan perintah [New-EC2Host](#) AWS Tools for Windows PowerShell . Perintah berikut mengalokasikan Host Khusus yang mendukung banyak tipe instans dari keluarga instans m5 di Zona Ketersediaan us-east-1a. Host juga mengaktifkan pemulihan host dan menonaktifkan penempatan otomatis.

```
PS C:\> New-EC2Host -InstanceFamily m5 -AvailabilityZone us-east-1a -
AutoPlacement Off -HostRecovery On -Quantity 1
```

Perintah berikut mengalokasikan Host Khusus yang mendukung peluncuran instans m4.large tidak tertarget di Zona Ketersediaan eu-west-1a, mengaktifkan pemulihan host, dan menerapkan tanda dengan kunci purpose dan nilai production.

Parameter TagSpecification yang digunakan untuk menandai Host Khusus saat pembuatan memerlukan objek yang menentukan tipe sumber daya yang akan diberi tanda, kunci tanda, dan nilai tanda. Perintah berikut membuat objek yang diperlukan.

```
PS C:\> $tag = @{ Key="purpose"; Value="production" }
PS C:\> $tagspec = new-object Amazon.EC2.Model.TagSpecification
PS C:\> $tagspec.ResourceType = "dedicated-host"
PS C:\> $tagspec.Tags.Add($tag)
```

Perintah berikut mengalokasikan Host Khusus dan menerapkan tanda yang ditentukan di objek `$tagspec`.

```
PS C:\> New-EC2Host -InstanceType m4.Large -AvailabilityZone eu-west-1a -  
AutoPlacement On -HostRecovery On -Quantity 1 -TagSpecification $tagspec
```

Luncurkan instans pada Host Khusus

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.

Tip

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](#).

Anda dapat meluncurkan sebuah instans ke Host Khusus menggunakan metode berikut.

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.

Note

AMI SQL Server, SUSE, dan RHEL yang disediakan oleh Amazon EC2 tidak dapat digunakan dengan Host Khusus.

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 Afinitas penghunian, lakukan salah satu hal berikut:
 - 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 [Pahami penempatan otomatis dan afinitas](#).

 Note

Opsi Penghunian dan Host telah dikonfigurasi sebelumnya berdasarkan host yang Anda pilih.

8. Konfigurasi opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
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.

Note

AMI SQL Server, SUSE, dan RHEL yang disediakan oleh Amazon EC2 tidak dapat digunakan dengan Host Khusus.

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 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 [Pahami penempatan otomatis dan afinitas](#).

7. Konfigurasi opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
8. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan sebuah instans ke Host Khusus

Gunakan AWS CLI perintah [run-instance](#) dan tentukan afinitas instance, tenancy, dan host dalam parameter permintaan. Placement

PowerShell

Untuk meluncurkan sebuah instans ke Host Khusus

Gunakan [New-EC2Instance](#) AWS Tools for Windows PowerShell perintah dan tentukan afinitas instance, penyewaan, dan host dalam parameter Placement permintaan.

Luncurkan sebuah instans ke dalam grup sumber daya host

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

- Anda harus mengaitkan konfigurasi lisensi berbasis inti atau soket dengan AMI.
- Anda tidak dapat menggunakan AMI 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.

Anda dapat meluncurkan sebuah instans ke dalam grup sumber daya host menggunakan metode berikut.

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.

Note

AMI SQL Server, SUSE, dan RHEL yang disediakan oleh Amazon EC2 tidak dapat digunakan dengan Host Khusus.

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 [Pahami penempatan otomatis dan afinitas](#).

7. Konfigurasi opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
8. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan sebuah instans ke dalam grup sumber daya host

Gunakan AWS CLI perintah [run-instance](#), dan dalam parameter Placement permintaan, hilangkan opsi Tenancy dan tentukan ARN grup sumber daya host.

PowerShell

Untuk meluncurkan sebuah instans ke dalam grup sumber daya host

Gunakan [New-EC2Instance](#) AWS Tools for Windows PowerShell perintah, dan dalam parameter Placement permintaan, hilangkan opsi Penyewaan dan tentukan grup sumber daya host ARN.

Pahami penempatan otomatis dan afinitas

Kontrol penempatan untuk Host Khusus terjadi pada level instans dan level host.

Penempatan otomatis

Penempatan otomatis dikonfigurasi di 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.

Jika penempatan otomatis Host Khusus dinonaktifkan, host hanya akan menerima peluncuran instans penghunian Host yang menentukan ID host uniknya. Ini adalah pengaturan default untuk Host Khusus baru.

Jika penempatan otomatis Host Khusus diaktifkan, host akan menerima semua peluncuran instans yang tidak ditargetkan yang cocok dengan konfigurasi tipe 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).

Memodifikasi penempatan otomatis Host Khusus

Anda dapat mengubah pengaturan penempatan otomatis Host Khusus setelah Anda mengalokasikannya ke AWS akun Anda, menggunakan salah satu metode berikut.

Console

Untuk mengubah 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 [Pahami penempatan otomatis dan afinitas](#).
5. Pilih Simpan.

AWS CLI

Untuk memodifikasi penempatan otomatis Host Khusus

Gunakan perintah [modify-hosts](#) AWS CLI . Contoh berikut memungkinkan penempatan otomatis untuk Host Khusus yang ditentukan.

```
aws ec2 modify-hosts --auto-placement on --host-ids h-012a3456b7890cdef
```

PowerShell

Untuk memodifikasi penempatan otomatis Host Khusus

Gunakan perintah [Edit-EC2Host](#) AWS Tools for Windows PowerShell . Contoh berikut memungkinkan penempatan otomatis untuk Host Khusus yang ditentukan.

```
PS C:\> Edit-EC2Host --AutoPlacement 1 --HostId h-012a3456b7890cdef
```

Mengubah tipe instans yang didukung

Dukungan untuk banyak tipe instans pada Host Khusus yang sama tersedia untuk keluarga instans berikut: C5, M5, R5, C5n, R5n, M5n, dan T3. Keluarga instans lain hanya mendukung satu tipe instans pada Host Khusus yang sama.

Anda dapat mengalokasikan Host Khusus menggunakan metode berikut ini.

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.

Anda dapat memodifikasi tipe instans yang didukung menggunakan salah satu metode berikut.

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 jenis instans yang didukung untuk Host Khusus

Gunakan perintah [modify-hosts](#) AWS CLI .

Perintah berikut mengubah Host Khusus untuk mendukung beberapa tipe instans dalam keluarga instans m5.

```
aws ec2 modify-hosts --instance-family m5 --host-ids h-012a3456b7890cdef
```

Perintah berikut mengubah Host Khusus untuk mendukung instans m5.xlarge 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 perintah [Edit-EC2Host](#) AWS Tools for Windows PowerShell .

Perintah berikut mengubah Host Khusus untuk mendukung beberapa tipe instans dalam keluarga instans m5.

```
PS C:\> Edit-EC2Host --InstanceFamily m5 --HostId h-012a3456b7890cdef
```

Perintah berikut memodifikasi Host Khusus untuk mendukung instans m5.xlarge saja.

```
PS C:\> Edit-EC2Host --InstanceType m5.xlarge --HostId h-012a3456b7890cdef
```

Memodifikasi penghunian dan afinitas instans

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`.

Anda dapat memodifikasi penghunian dan afinitas sebuah instans menggunakan metode berikut.

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 Modify instance placement, konfigurasi hal berikut:
 - Penghunian—Pilih salah satu dari berikut:
 - Jalankan instans perangkat keras khusus — Meluncurkan instans sebagai Instans Khusus. Untuk informasi selengkapnya, lihat [Instans Khusus](#).
 - 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 [Pahami penempatan otomatis dan afinitas](#).

6. Pilih Simpan.

AWS CLI

Untuk mengubah tenancy atau afinitas instance

Gunakan perintah [AWS CLI 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 perintah [Edit-EC2InstancePlacement](#) AWS Tools for Windows PowerShell . Contoh berikut mengubah afinitas instans yang ditetapkan dari default menjadi host, dan menetapkan Host Khusus yang mempunyai afinitas dengan instans.

```
PS C:\> Edit-EC2InstancePlacement -InstanceId i-1234567890abcdef0 -Affinity host -Tenancy host -HostId h-012a3456b7890cdef
```

Melihat Host Khusus

Anda dapat melihat detail tentang Host Khusus dan masing-masing instans di dalamnya menggunakan metode berikut.

Console

Untuk melihat detail dari 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.
4. Untuk informasi tentang host, pilih Detail.

vCPU yang Tersedia menunjukkan vCPU yang tersedia di Host Khusus untuk peluncuran instans baru. Sebagai contoh, Host Khusus yang mendukung banyak tipe instans dalam keluarga instans c5, dan yang tidak memiliki instans berjalan di atasnya, memiliki 72 vCPU yang tersedia. Hal ini berarti bahwa Anda dapat meluncurkan kombinasi tipe instans yang berbeda ke Host Khusus untuk menggunakan 72 vCPU yang tersedia.

Untuk informasi tentang instans yang berjalan di host, pilih Menjalankan instans.

AWS CLI

Untuk melihat kapasitas Host Khusus

Gunakan [perintah describe-hosts](#) AWS CLI .

Contoh berikut menggunakan perintah [describe-hosts](#) (AWS CLI) untuk melihat kapasitas instans yang tersedia untuk Host Khusus yang mendukung beberapa tipe instans dalam keluarga instans c5. Host Khusus sudah memiliki dua instans c5.4xlarge dan empat instans c5.2xlarge yang berjalan di atasnya.

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

```
"AvailableInstanceCapacity": [  
  { "AvailableCapacity": 2,  
    "InstanceType": "c5.xlarge",  
    "TotalCapacity": 18 },  
  { "AvailableCapacity": 4,  
    "InstanceType": "c5.large",  
    "TotalCapacity": 36 }  
],  
"AvailableVCpus": 8
```

PowerShell

Untuk melihat kapasitas instans dari Host Khusus

Gunakan perintah [Get-EC2Host](#) AWS Tools for Windows PowerShell .

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

Tandai Host Khusus

Anda dapat menetapkan tanda kustom ke Host Khusus yang ada untuk mengategorikannya dengan cara berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Ini membantu Anda untuk menemukan Host Khusus dengan cepat berdasarkan tanda kustom yang Anda tetapkan. Tanda Host Khusus juga dapat digunakan untuk pelacakan alokasi biaya.

Anda juga dapat menerapkan tanda ke Host Khusus pada saat pembuatan. Untuk informasi selengkapnya, lihat [Alokasikan Host Khusus](#).

Anda dapat mengalokasikan Host Khusus menggunakan metode berikut ini.

Console

Untuk menandai 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 ditandai, lalu pilih Tindakan, Kelola tanda.
4. Di layar Kelola tanda, pilih Tambahkan tanda, lalu tentukan kunci dan nilai untuk tanda tersebut.
5. (Opsional) Pilih Tambahkan tanda untuk menambahkan tanda tambahan ke Host Khusus.
6. Pilih Simpan Perubahan.

AWS CLI

Untuk menandai Host Khusus

Gunakan perintah [create-tags](#) AWS CLI .

Perintah berikut menandai Host Khusus dengan Owner=TeamA.

```
aws ec2 create-tags --resources h-abc12345678909876 --tags Key=Owner,Value=TeamA
```

PowerShell

Untuk menandai Host Khusus

Gunakan perintah [New-EC2Tag](#) AWS Tools for Windows PowerShell .

Perintah `New-EC2Tag` memerlukan objek `Tag`, yang menentukan pasangan kunci dan nilai yang akan digunakan untuk tanda Reservasi Host Khusus. Perintah berikut membuat objek `Tag` bernama `$tag`, dengan pasangan kunci dan nilai `Owner` dan `TeamA`.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

Perintah berikut menandai Host Khusus yang ditentukan dengan objek `$tag`.

```
PS C:\> New-EC2Tag -Resource h-abc12345678909876 -Tag $tag
```

Memantau Host Khusus

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 AWS CLI perintah [describe-hosts](#) dan kemudian tinjau `state` properti di elemen respon. `hostSet`

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

PowerShell

Untuk melihat status Host Khusus

Gunakan [Get-EC2Host](#) AWS Tools for Windows PowerShell perintah dan kemudian tinjau state properti di elemen `hostSet` respons.

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

Tabel berikut menjelaskan kemungkinan status Host Khusus.

Status	Deskripsi
<code>available</code>	AWS belum mendeteksi masalah dengan Host Khusus. Tidak ada pemeliharaan atau perbaikan yang dijadwalkan. Instans dapat diluncurkan ke Host Khusus ini.
<code>released</code>	Host Khusus telah dilepas. ID host tidak lagi digunakan. Host yang dilepas tidak dapat digunakan kembali.
<code>under-assessment</code>	AWS sedang mengeksplorasi kemungkinan masalah dengan Host Khusus. Jika tindakan harus diambil, Anda akan diberitahu melalui AWS Management Console atau email. Instans tidak dapat diluncurkan ke Host Khusus dalam status ini.
<code>pending</code>	Host Khusus tidak dapat digunakan untuk peluncuran instans baru. Modifikasi untuk mendukung banyak tipe instans atau pemulihan host sedang berlangsung.
<code>permanent-failure</code>	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 pada Host Khusus dengan status ini, host akan AWS pensiun. AWS tidak memulai ulang instance dalam keadaan ini. Instans tidak dapat diluncurkan ke Host Khusus dalam status ini.
<code>released-permanent-failure</code>	AWS secara permanen merilis Host Khusus yang gagal dan tidak lagi menjalankan instance di dalamnya. ID Host Khusus tidak lagi tersedia untuk digunakan.

Melepas Host Khusus

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 Host Khusus Sesuai Permintaan.

Anda dapat melepas Host Khusus menggunakan metode berikut ini.

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 CLI .

```
aws ec2 release-hosts --host-ids h-012a3456b7890cdef
```

PowerShell

Untuk merilis Host Khusus

Gunakan perintah [Remove-EC2Hosts](#) AWS Tools for Windows PowerShell .

```
PS C:\> Remove-EC2Hosts -HostId h-012a3456b7890cdef
```

Setelah Anda merilis Host Khusus, Anda tidak dapat menggunakan kembali host atau ID host yang sama, sehingga Anda tidak lagi dikenai tarif penagihan Sesuai Permintaan. 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.

Membeli Reservasi Host Khusus

Anda dapat membeli reservasi menggunakan metode berikut:

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 Instans, pilih keluarga instans dari Host Khusus untuk membeli Reservasi Tuan Rumah Khusus.
 - b. Untuk opsi Pembayaran, pilih dan konfigurasi opsi pembayaran pilihan Anda.
4. Pilih Selanjutnya.
5. Pilih Host Khusus untuk mengaitkan Reservasi Tuan Rumah Khusus, lalu pilih Berikutnya.
6. (Opsional) Tetapkan tag ke Reservasi Tuan Rumah Khusus.
7. Tinjau pesanan Anda dan pilih Pembelian.

AWS CLI

Untuk membeli reservasi

1. Gunakan [perintah `describe-host-reservation-offerings` untuk membuat daftar penawaran](#) yang tersedia yang sesuai dengan AWS CLI kebutuhan Anda. Contoh berikut menampilkan daftar

penawaran yang mendukung instans di keluarga instans m4 dan memiliki jangka waktu satu tahun.

 Note

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. Perhatikan `offeringId` dari penawaran yang akan dibeli.

- Gunakan AWS CLI perintah [pembeli-host-reservasi](#) untuk membeli penawaran dan berikan yang `offeringId` disebutkan di 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

- Gunakan [Get-EC2HostReservationOffering](#) AWS Tools for Windows PowerShell perintah untuk membuat daftar penawaran yang tersedia yang sesuai dengan kebutuhan Anda. Contoh berikut mencantumkan penawaran yang mendukung instans di keluarga instans m4 dan memiliki jangka waktu satu tahun.

 Note

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.

```
PS C:\> $filter = @{"Name"="instance-family"; Value="m4"}
```

```
PS C:\> Get-EC2HostReservationOffering -filter $filter -MaxDuration 31536000
```

Perintah menampilkan daftar penawaran yang sesuai dengan kriteria Anda. Perhatikan `offeringId` dari penawaran yang akan dibeli.

- Gunakan [New-EC2HostReservation](#) AWS Tools for Windows PowerShell perintah untuk membeli penawaran dan berikan yang `offeringId` disebutkan pada langkah sebelumnya. Contoh berikut membeli reservasi yang ditentukan dan mengaitkannya dengan Host Khusus tertentu yang sudah dialokasikan di AWS akun.

```
PS C:\> New-EC2HostReservation -OfferingId hro-03f707bf363b6b324 -  
HostIdSet h-013abcd2a00cbd123
```

Melihat reservasi Host Khusus

Anda dapat melihat informasi tentang Host Khusus yang terkait dengan reservasi Anda, termasuk:

- Jangka waktu reservasi
- Opsi pembayaran
- Tanggal mulai dan berakhir

Anda dapat melihat detail reservasi Host Khusus menggunakan metode berikut.

Console

Untuk melihat detail reservasi Host Khusus

- Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
- Pilih Host Khusus di panel navigasi.
- Di halaman Host Khusus, pilih Reservasi Host Khusus, lalu pilih reservasi dari daftar yang disediakan.
- Pilih Detail untuk informasi tentang reservasi.
- Pilih Host untuk informasi tentang Host Khusus yang terkait dengan reservasi.

AWS CLI

Untuk melihat detail reservasi Host Khusus

Gunakan [perintah deskripsi-host-reservations](#) AWS CLI .

```
aws ec2 describe-host-reservations
```

PowerShell

Untuk melihat detail reservasi Host Khusus

Gunakan perintah [Get-EC2HostReservation](#) AWS Tools for Windows PowerShell .

```
PS C:\> Get-EC2HostReservation
```

Menandai Reservasi Host Khusus

Anda dapat menetapkan tanda kustom ke Reservasi Host Khusus untuk mengategorikannya dengan cara berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Ini membantu Anda untuk menemukan Reservasi Host Khusus dengan cepat berdasarkan tanda kustom yang Anda tetapkan.

Anda dapat menandai Reservasi Host Khusus menggunakan alat baris perintah saja.

AWS CLI

Untuk menandai Reservasi Host Khusus

Gunakan perintah [create-tags](#) AWS CLI .

```
aws ec2 create-tags --resources hr-1234563a4ffc669ae --tags Key=Owner,Value=TeamA
```

PowerShell

Untuk menandai Reservasi Host Khusus

Gunakan perintah [New-EC2Tag](#) AWS Tools for Windows PowerShell .

Perintah New-EC2Tag memerlukan parameter Tag, yang menentukan pasangan kunci dan nilai untuk digunakan untuk tanda Reservasi Host Khusus. Perintah berikut akan membuat parameter Tag.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource hr-1234563a4ffc669ae -Tag $tag
```

Bekerja dengan Host Khusus bersama

Berbagi Host Khusus memungkinkan pemilik Host Khusus untuk berbagi Host Khusus mereka dengan AWS akun lain atau di dalam AWS organisasi. Hal 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 spesifik 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](#)
- [Berbagi Host Khusus](#)

- [Batalkan berbagi Host Khusus bersama](#)
- [Mengidentifikasi Host Khusus bersama](#)
- [Tampilkan instans yang berjalan pada Host Khusus bersama](#)
- [Izin Host Khusus Bersama](#)
- [Tagihan dan pengukuran](#)
- [Batas Host Khusus](#)
- [Pemulihan host dan berbagi Host Khusus](#)

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 Anda 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 pembagian 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 selengkapnya AWS RAM, 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, Availability Zone us-east-1a untuk AWS akun Anda mungkin tidak memiliki lokasi yang sama dengan AWS akun lain. us-east-1a

Untuk mengidentifikasi lokasi Host Khusus Anda yang terkait dengan akun Anda, Anda harus menggunakan ID Zona Ketersediaan (AZ ID). ID Availability Zone adalah pengidentifikasi unik dan konsisten untuk Availability Zone 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>.
2. ID Zona Ketersediaan untuk Wilayah saat ini ditampilkan di panel ID AZ Anda di sisi kanan layar.

Berbagi Host Khusus

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.

Anda dapat berbagi Host Khusus yang Anda miliki dengan menggunakan salah satu dari metode berikut ini.

Amazon EC2 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.

AWS RAM console

Untuk berbagi Host Khusus yang Anda miliki menggunakan AWS RAM konsol

Lihat [Membuat Sumber Daya Bersama](#) di Panduan Pengguna AWS RAM .

AWS CLI

Untuk berbagi Host Khusus yang Anda miliki menggunakan AWS CLI

Gunakan perintah [create-resource-share](#).

Batalkan berbagi Host Khusus bersama

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. Anda dapat melakukan ini dengan menggunakan salah satu metode berikut.

Amazon EC2 console

Untuk membatalkan berbagi 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 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.

AWS RAM console

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki menggunakan konsol AWS RAM

Lihat [Memperbarui Sumber Daya Bersama](#) di Panduan Pengguna AWS RAM .

Command line

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki menggunakan AWS CLI

Gunakan perintah [disassociate-resource-share](#).

Mengidentifikasi Host Khusus bersama

Pemilik dan konsumen dapat mengidentifikasi Host Khusus bersama menggunakan salah satu metode berikut.

Amazon EC2 console

Untuk mengidentifikasi Host Khusus bersama menggunakan konsol Amazon EC2

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. Kolom Pemilik menunjukkan ID akun AWS dari pemilik Host Khusus.

Command line

Untuk mengidentifikasi Host Khusus bersama menggunakan AWS CLI

Gunakan perintah [describe-host](#). Perintah tersebut menampilkan Host Khusus yang Anda miliki dan Host Khusus yang dibagikan dengan Anda.

Tampilkan instans yang berjalan pada Host Khusus bersama

Pemilik dan konsumen dapat melihat instans yang berjalan pada Host Khusus bersama kapan saja menggunakan salah satu metode berikut.

Amazon EC2 console

Untuk melihat instans yang berjalan pada Host Khusus bersama 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 melihat instans dan pilih Instans. Tab ini menampilkan daftar instans yang berjalan di host. Pemilik melihat semua instans yang berjalan di host, termasuk instans yang diluncurkan oleh konsumen. Konsumen hanya melihat instans berjalan yang mereka luncurkan ke host. Kolom Pemilik menunjukkan ID akun AWS dari akun yang meluncurkan instans.

Command line

Untuk melihat instans yang berjalan pada Host Khusus bersama menggunakan AWS CLI

Gunakan perintah [describe-host](#). Perintah tersebut mengembalikan instans yang berjalan di setiap Host Khusus. Pemilik melihat semua instans yang berjalan di host. Konsumen hanya melihat instans berjalan yang mereka luncurkan di host bersama. InstanceOwnerId menunjukkan ID akun AWS dari pemilik instans.

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](#).

Tuan Rumah Khusus di 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 untuk membangun dan menjalankan aplikasi di tempat menggunakan antarmuka pemrograman yang sama seperti di AWS Wilayah, 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 suatu AWS Wilayah.

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 dan gunakan Host Khusus di Outpost](#)

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 instans berikut didukung: C5, M5, R5, C5d, M5d, R5d, G4dn, dan i3en.
- Host Khusus di Outposts dapat dikonfigurasi untuk mendukung beberapa ukuran instans. Dukungan untuk banyak ukuran instans tersedia untuk keluarga instans berikut: C5, M5, R5, C5d, M5d, dan R5d. Untuk informasi selengkapnya, lihat [Konfigurasi kapasitas instans](#).
- Host Khusus di Outposts mendukung penempatan otomatis dan peluncuran instans tertarget. Untuk informasi selengkapnya, lihat [Pahami penempatan otomatis dan afinitas](#).
- Host Khusus di Outposts mendukung afinitas host. Untuk informasi selengkapnya, lihat [Pahami penempatan otomatis dan afinitas](#).
- Host Khusus di Outposts mendukung berbagi dengan. AWS RAM Untuk informasi selengkapnya, lihat [Bekerja dengan Host Khusus bersama](#).

Pertimbangan

- Reservasi Host Khusus tidak didukung di Outposts.
- Host grup sumber daya dan tidak AWS License Manager didukung di Outposts.
- 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 instance dengan penyewaan Host Khusus di Outposts.

Alokasikan dan gunakan Host Khusus di Outpost

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:

AWS Outposts console

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. Konfigurasi Host Khusus sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Alokasikan Host Khusus](#).

Note

Zona Ketersediaan dan ARN Outpost harus diisi sebelumnya dengan Zone Ketersediaan dan ARN dari Outpost yang dipilih.

4. Pilih Alokasikan.

Amazon EC2 console

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. Konfigurasi pengaturan Host Khusus yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Alokasikan Host Khusus](#).
7. Pilih Alokasikan.

AWS CLI

Gunakan perintah [allocate-hosts](#) AWS CLI . 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
```

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. Konfigurasi instans sesuai kebutuhan kemudian luncurkan instans. Untuk informasi selengkapnya, lihat [Luncurkan instans pada Host Khusus](#).

Pemulihan host

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

- [Dasar-dasar pemulihan host](#)
- [Tipe instans yang didukung](#)
- [Konfigurasi pemulihan host](#)
- [Status pemulihan host](#)
- [Memulihkan secara manual instans yang tidak didukung](#)
- [Layanan-layanan terkait](#)
- [Penetapan harga](#)

Dasar-dasar pemulihan host

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 daya sistem atau kegagalan 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 AWS License Manager mengotomatiskan pelacakan dan pengelolaan lisensi Anda.

 Note

AWS Integrasi License Manager hanya didukung di Wilayah di mana AWS License Manager 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.

Ketika pemulihan host dimulai, pemilik AWS akun diberitahu melalui email dan oleh suatu AWS Health Dashboard acara. Notifikasi kedua dikirimkan setelah pemulihan host berhasil diselesaikan.

Jika Anda menggunakan AWS License Manager untuk melacak lisensi Anda, AWS License Manager mengalokasikan lisensi baru untuk penggantian Host Khusus berdasarkan batas konfigurasi lisensi. Jika konfigurasi lisensi memiliki batas keras yang akan dilanggar sebagai akibat dari pemulihan host, proses pemulihan tidak diperbolehkan dan Anda diberitahu tentang kegagalan pemulihan host melalui pemberitahuan Amazon SNS (jika pengaturan pemberitahuan telah dikonfigurasi

untuk License AWS Manager). 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.

Skenario tanpa pemulihan otomatis Host Khusus

Pemulihan otomatis Host Khusus tidak terjadi ketika host dijadwalkan pensiun. Anda akan menerima pemberitahuan pensiun di AWS Health Dashboard 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.

Tipe instans yang didukung

Pemulihan host didukung untuk keluarga contoh berikut: A1, C3, C4, C5, C5n, C6a, C6g, C6i, Inf1, G3, G5g, M3, M4, M5, M5n, M5zn, M6a, M6g, M6i, P2, P3, R3, R4, R5, R5B, R5n, R6g, R6i, T3, X1, X1e, X2IEZn, u-6tb1, u-9tb1, u-12tb1, u-18tb1, dan u-24tb1.

Untuk memulihkan instans yang tidak didukung, lihat [Memulihkan secara manual instans yang tidak didukung](#).

Note

Pemulihan otomatis Host khusus untuk tipe instans metal yang didukung akan membutuhkan waktu lebih lama untuk mendeteksi dan memulihkan dari tipe instans nonmetal.

Konfigurasi pemulihan host

Anda dapat mengonfigurasi pemulihan host pada saat alokasi Host Khusus, atau setelah alokasi menggunakan konsol Amazon EC2 atau AWS Command Line Interface (CLI).

Daftar Isi

- [Aktifkan pemulihan host](#)
- [Nonaktifkan pemulihan host](#)
- [Tampilkan konfigurasi pemulihan host](#)

Aktifkan pemulihan host

Anda dapat mengaktifkan pemulihan host pada saat alokasi Host Khusus atau setelah alokasi.

Untuk informasi lebih lanjut tentang mengaktifkan pemulihan host pada saat alokasi Host Khusus, lihat [Alokasikan Host Khusus](#).

Untuk menonaktifkan pemulihan host setelah alokasi menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Khusus untuk mengaktifkan pemulihan host, lalu pilih Tindakan, Ubah Pemulihan Host.
4. Untuk Pemulihan host, pilih Aktifkan, lalu pilih Simpan.

Untuk mengaktifkan pemulihan host setelah alokasi menggunakan AWS CLI

Gunakan perintah [modify-host](#) dan tentukan parameter `host-recovery`.

```
$ aws ec2 modify-hosts --host-recovery on --host-ids h-012a3456b7890cdef
```

Nonaktifkan pemulihan host

Anda dapat menonaktifkan pemulihan host kapan saja setelah Host Khusus dialokasikan.

Untuk menonaktifkan pemulihan host setelah alokasi menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Khusus yang akan menonaktifkan pemulihan host, lalu pilih Tindakan, Ubah Pemulihan Host.
4. Untuk Pemulihan host, pilih Nonaktifkan, lalu pilih Simpan.

Untuk menonaktifkan pemulihan host setelah alokasi menggunakan AWS CLI

Gunakan perintah [modify-host](#) dan tentukan parameter `host-recovery`.

```
$ aws ec2 modify-hosts --host-recovery off --host-ids h-012a3456b7890cdef
```

Tampilkan konfigurasi pemulihan host

Anda dapat melihat konfigurasi pemulihan host untuk Host Khusus kapan saja.

Untuk melihat konfigurasi pemulihan host untuk Host Khusus menggunakan konsol

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.

Untuk melihat konfigurasi pemulihan host untuk Host Khusus menggunakan AWS CLI

Gunakan perintah [describe-host](#).

```
$ aws ec2 describe-hosts --host-ids h-012a3456b7890cdef
```

Elemen respons `HostRecovery` menunjukkan apakah pemulihan host diaktifkan atau dinonaktifkan.

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 mengalami gangguan memiliki instans yang tidak mendukung pemulihan host, seperti instans dengan volume yang didukung penyimpanan instans, Host Khusus tidak akan dirilis. Sebaliknya, host ditandai untuk pensiun dan memasuki status `permanent-failure`.

Memulihkan secara manual instans yang tidak didukung

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. Termasuk di dalamnya volume penyimpanan instans yang dilampirkan ke instans yang memiliki volume EBS sebagai perangkat root. Untuk melindungi data dari volume penyimpanan instans, cadangkan ke penyimpanan persisten sebelum instans dihentikan atau diakhiri.

Memulihkan instans yang didukung EBS secara manual

Untuk instans yang didukung EBS yang tidak dapat dipulihkan secara otomatis, kami menyarankan Anda untuk menghentikan dan memulai instans secara manual untuk memulihkannya ke Host Khusus yang baru. Untuk informasi selengkapnya tentang menghentikan instans Anda, dan tentang perubahan yang terjadi dalam konfigurasi instans Anda saat dihentikan, lihat [Hentikan dan mulai instans Amazon EC2](#).

Pulihkan instans yang didukung penyimpanan instans secara manual

Untuk instans yang didukung penyimpanan instans yang tidak dapat dipulihkan secara otomatis, kami menyarankan Anda untuk melakukan 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.

Layanan-layanan terkait

Host Khusus terintegrasi dengan layanan berikut:

- AWS License Manager —Melacak lisensi di seluruh Host Khusus Amazon EC2 Anda (hanya didukung di Wilayah di mana License AWS Manager tersedia). Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS License Manager](#).

Penetapan 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 terganggu ditagih menggunakan tarif Sesuai Permintaan, Host Khusus pengganti juga akan ditagih menggunakan tarif Sesuai Permintaan. Jika Host Khusus yang mengalami gangguan memiliki Reservasi Host Khusus yang aktif, Host Khusus tersebut akan ditransfer ke Host Khusus pengganti.

Pemeliharaan host

Dengan pemeliharaan host, instans Amazon EC2 Anda pada Host Khusus yang terdegradasi secara otomatis di-boot ulang pada Host Khusus pengganti selama acara pemeliharaan terjadwal. Hal ini membantu mengurangi waktu henti aplikasi dan memindahkan beban pemeliharaan berat yang tidak terdiferensiasi ke AWS. Pemeliharaan host juga dilakukan untuk pemeliharaan Amazon EC2 terencana dan rutin.

Pemeliharaan host didukung pada semua alokasi Host Khusus baru yang dibuat melalui konsol Amazon EC2. Untuk Host Khusus apa pun di Host Khusus Anda Akun AWS atau Host Terdedikasi baru yang dialokasikan melalui [AllocateHosts](#) API, Anda dapat mengonfigurasi pemeliharaan host untuk Host Khusus yang didukung. Untuk informasi selengkapnya, lihat [the section called "Mengonfigurasi pemeliharaan host"](#).

Daftar Isi

- [Dasar-dasar pemeliharaan host](#)
- [Pemeliharaan host versus pemulihan host](#)
- [Tipe instans yang didukung](#)
- [Instans pada Host Khusus](#)
- [Mengonfigurasi pemeliharaan host](#)
- [Peristiwa pemeliharaan](#)
- [Status pemeliharaan host](#)
- [Layanan terkait](#)
- [Penetapan harga](#)

Dasar-dasar pemeliharaan host

Ketika degradasi terdeteksi pada Host Khusus, Host Khusus baru dialokasikan. Degradasi dapat disebabkan oleh degradasi perangkat keras dasar atau deteksi kondisi bermasalah tertentu. Instans Anda pada Host Khusus yang terdegradasi dijadwalkan untuk di-boot ulang secara otomatis pada Host Khusus pengganti.

Host Khusus pengganti menerima ID host baru, tetapi mempertahankan atribut yang sama dengan Host Khusus yang asli. Atribut ini meliputi yang berikut ini.

- Pengaturan penempatan otomatis
- Zona Ketersediaan
- Reservasi
- Afinitas host
- Pengaturan pemeliharaan host
- Pengaturan pemulihan host
- Jenis instans
- Tanda

Pemeliharaan host tersedia di semua Wilayah AWS untuk semua Host Khusus yang didukung. Untuk informasi selengkapnya tentang Host Khusus yang tidak mendukung pemeliharaan host, lihat [the section called “Batasan”](#).

Host Khusus Anda yang terdegradasi dirilis setelah semua instans Anda di-boot ulang ke Host Khusus baru atau dihentikan. Anda dapat mengakses instans Anda di Host Khusus yang terdegradasi sebelum peristiwa pemeliharaan terjadwal, tetapi peluncuran instans pada Host Khusus yang terdegradasi tidak didukung.

Anda dapat menggunakan Host Khusus pengganti untuk meluncurkan instans baru di host sebelum acara pemeliharaan terjadwal. Namun, beberapa kapasitas instance pada host pengganti dicadangkan untuk instance yang harus dimigrasikan dari host yang terdegradasi. Anda tidak dapat meluncurkan instans baru ke dalam kapasitas cadangan ini. Untuk informasi selengkapnya, lihat [the section called “Instans pada Host Khusus”](#).

Batasan

- Pemeliharaan host tidak didukung di AWS Outposts, AWS Local Zones, dan AWS Wavelength Zones.
- Pemeliharaan host tidak dapat diaktifkan atau dinonaktifkan untuk host yang sudah ada 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 hanya didukung pada tipe instans tertentu. Untuk informasi selengkapnya, lihat [the section called “Tipe instans yang didukung”](#).

Pemeliharaan host versus pemulihan host

Tabel berikut menunjukkan perbedaan utama antara pemulihan host dan pemeliharaan host.

	Pemulihan host	Pemeliharaan host
Aksesibilitas	Tidak dapat dijangkau	Dapat dijangkau
Status	<code>under-assessment</code>	<code>permanent-failure</code>
Tindakan	Pemulihan segera dilakukan	Pemeliharaan dijadwalkan
Fleksibilitas penjadwalan	Tidak dapat dijadwalkan ulang	Dapat dijadwalkan ulang
Grup Sumber Daya Host	Didukung	Tidak didukung

Untuk informasi selengkapnya tentang pemulihan host, lihat [Pemulihan host](#).

Tipe instans yang didukung

Pemeliharaan host didukung untuk keluarga instans berikut:

- Tujuan umum: A1 M4 | M5 | M5a M5n | M5zn | M6a | M6g | M6i | M6in M7a | M7g | M7i | T3
- Komputasi yang dioptimalkan: C4 | C5 | C5a | C5n | C6a | C6g | C6gn | C6i | C6in | C7g | C7gn | C7i
- Memori yang dioptimalkan: R4 | R5 | R5a | R5b | R5n | R6a | R6g | R6i | R6in | R7a | R7g | R7iz | u-12tb1 | u-18tb1 | u-24tb1 | u-3tb1 | u-6tb1 | u-9tb1 | X2iezn
- Komputasi yang dipercepat: G3 | G5g | Inf1 | P2 | P3

Instans pada Host Khusus

Amazon EC2 secara otomatis menyimpan kapasitas pada host pengganti untuk instans yang akan dimigrasi secara otomatis dari host yang terdegradasi. Amazon EC2 tidak menyimpan kapasitas pada host pengganti untuk instans yang tidak dapat dimigrasi secara otomatis, seperti instance dengan volume root penyimpanan instans. Kapasitas cadangan tidak dapat digunakan untuk meluncurkan instans baru.

Note

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 hingga dihentikan atau dimigrasikan ke kapasitas cadangan pada host pengganti.

Jika Anda menghentikan instance secara manual pada host terdegradasi yang dapat dimigrasi secara otomatis, kapasitas yang dicadangkan untuk instance tersebut di host pengganti akan dirilis dan tersedia untuk digunakan.

Selama acara pemeliharaan terjadwal, instans pada host terdegradasi di-boot ulang dan dimigrasikan ke kapasitas cadangan pada Host Khusus pengganti. Instans yang dimigrasi mempertahankan atribut yang sama dengan yang ada di host terdegradasi Anda, termasuk yang berikut ini.

- Lampiran volume Amazon EBS
- Alamat IP elastic
- ID Instans

- Metadata instans
- Alamat IP privat

Anda dapat menghentikan dan memulai instans pada host yang terdegradasi kapan saja sebelum peristiwa pemeliharaan terjadwal dimulai. Melakukan hal ini akan menyebabkan boot ulang instans Anda ke host lain, dan instans Anda tidak akan menjalani pemeliharaan terjadwal. Anda harus memperbarui afinitas host instans Anda ke host baru tempat Anda ingin melakukan boot ulang instans Anda. Jika Anda menghentikan semua instance pada host yang terdegradasi sebelum acara pemeliharaan dimulai, host yang terdegradasi akan dirilis dan acara pemeliharaan dibatalkan. Untuk informasi selengkapnya, lihat [Hentikan dan mulai instans Amazon EC2](#).

Note

Data volume penyimpanan lokal apa pun tidak disimpan saat Anda menghentikan dan memulai instans Anda.

Instans dengan volume penyimpanan instans sebagai perangkat root diakhiri setelah tanggal penghentian yang ditentukan. Data apa pun pada volume penyimpanan instans dihapus ketika instans gagal atau berakhir. Instans yang diakhiri dihapus secara permanen, dan tidak dapat dimulai lagi. Untuk instans dengan volume penyimpanan instans sebagai perangkat root, sebaiknya luncurkan instans pengganti pada Host Khusus yang berbeda menggunakan Amazon Machine Image terbaru, dan migrasikan semua data yang tersedia ke instans pengganti sebelum tanggal pengakhiran yang ditentukan. Untuk informasi lebih lanjut, lihat [Tindakan yang harus diambil misalnya pensiun](#).

Instans yang tidak dapat di-boot ulang secara otomatis dihentikan setelah tanggal yang ditentukan. Anda dapat memulai instans ini lagi di host yang berbeda. Instans yang menggunakan volume Amazon EBS sebagai perangkat root terus menggunakan volume Amazon EBS yang sama setelah dimulai pada host baru.

Anda dapat mengatur urutan boot ulang instans dengan menjadwalkan ulang waktu mulai boot ulang instans di <https://console.aws.amazon.com/ec2/>.

Mengonfigurasi pemeliharaan host

Anda dapat mengonfigurasi pemeliharaan host untuk semua Host Khusus yang didukung melalui AWS Management Console atau AWS CLI. Lihat tabel berikut untuk detail selengkapnya.

AWS Management Console

Untuk mengaktifkan pemeliharaan host untuk Host Khusus Anda menggunakan AWS Management Console.

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 menggunakan AWS Management Console.

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.

Untuk melihat konfigurasi pemeliharaan host untuk Host Khusus Anda menggunakan AWS Management Console.

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 Pemeliharaan host.

AWS CLI

Untuk mengaktifkan atau menonaktifkan pemeliharaan host untuk Host Khusus Anda selama alokasi menggunakan AWS CLI.

Gunakan perintah [allocate-hosts](#).

Aktifkan

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --availability-zone us-east-1b --host-maintenance on
```

Nonaktifkan

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --availability-zone us-east-1b --host-maintenance off
```

Untuk mengaktifkan atau menonaktifkan pemeliharaan host untuk Host Khusus yang ada menggunakan AWS CLI.

Gunakan perintah [modify-hosts](#).

Aktifkan

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance on --host-ids h-0d123456bbf78910d
```

Nonaktifkan

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance off --host-ids h-0d123456bbf78910d
```

Untuk melihat konfigurasi pemeliharaan host untuk Host Khusus Anda menggunakan AWS CLI.

Gunakan perintah [describe-host](#).

```
aws ec2 describe-hosts --region us-east-1 --host-ids h-0d123456bbf78910d
```

Note

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.

Peristiwa pemeliharaan

Saat mendeteksi degradasi, peristiwa pemeliharaan dijadwalkan 14 hari kemudian, untuk melakukan boot ulang instans Anda di Host Khusus baru. Anda menerima notifikasi email yang

memberikan detail tentang host yang terdegradasi, peristiwa pemeliharaan terjadwal, dan slot waktu pemeliharaan. Untuk informasi selengkapnya, lihat [Melihat peristiwa terjadwal](#).

Anda dapat menjadwalkan ulang peristiwa pemeliharaan untuk hari apa pun hingga tujuh hari setelah tanggal peristiwa yang dijadwalkan. Untuk informasi selengkapnya tentang penjadwalan ulang, lihat [Menjadwalkan ulang peristiwa yang dijadwalkan](#).

Peristiwa pemeliharaan biasanya memakan waktu beberapa menit. Dalam peristiwa gagal yang jarang terjadi, Anda menerima notifikasi email untuk mengusir instans pada host terdegradasi dalam jangka waktu tertentu.

Status pemeliharaan host

Host Khusus Anda diatur ke status `permanent-failure` saat degradasi terdeteksi. Anda tidak dapat meluncurkan instans pada Host Khusus dalam status `permanent-failure`. Setelah menyelesaikan peristiwa pemeliharaan, host yang terdegradasi dilepaskan dan dimasukkan ke dalam status `released`, `permanent-failure`.

Setelah mendeteksi degradasi pada Host Khusus dan sebelum menjadwalkan acara pemeliharaan, pemeliharaan host secara otomatis mengalokasikan Host Khusus pengganti di akun Anda. Tuan rumah pengganti ini tetap dalam pending keadaan sampai acara pemeliharaan dijadwalkan. Setelah acara pemeliharaan dijadwalkan, Host Khusus pengganti pindah ke `available` negara bagian.

Anda dapat menggunakan Host Khusus pengganti untuk meluncurkan instans baru di host sebelum acara pemeliharaan terjadwal. Namun, beberapa kapasitas instance pada host pengganti dicadangkan untuk instance yang harus dimigrasikan dari host yang terdegradasi. Anda tidak dapat meluncurkan instans baru ke dalam kapasitas cadangan ini. Untuk informasi selengkapnya, lihat [the section called "Instans pada Host Khusus"](#).

Layanan terkait

Host Khusus terintegrasi dengan AWS License Manager —Melacak lisensi di seluruh Host Khusus Amazon EC2 Anda (hanya didukung di Wilayah di mana AWS License Manager tersedia). Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS License Manager](#).

Anda harus memiliki lisensi yang cukup Akun AWS untuk Host Khusus Anda yang baru. Lisensi yang terkait dengan host terdegradasi Anda dirilis saat host dilepas setelah selesainya peristiwa pemeliharaan terjadwal.

Penetapan 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 Dedicated Host yang terdegradasi ditagih menggunakan tarif On-Demand, Dedicated Host pengganti juga ditagih menggunakan tarif On-Demand. Jika Host Khusus yang terdegradasi memiliki Reservasi Host Khusus yang aktif, Host Khusus tersebut akan ditransfer ke Host Khusus yang baru.

Lacak perubahan konfigurasi

Anda dapat menggunakan AWS Config untuk merekam perubahan konfigurasi untuk Host Khusus, dan untuk instance yang diluncurkan, dihentikan, atau dihentikan pada mereka. Anda kemudian dapat menggunakan informasi yang ditangkap oleh AWS Config sebagai sumber data untuk pelaporan lisensi.

AWS Config mencatat informasi konfigurasi untuk Host Khusus dan instans satu per satu, dan memasang informasi ini melalui hubungan. Ada tiga kondisi pelaporan:

- **AWS Config status perekaman**—Saat Aktif, AWS Config merekam satu atau beberapa 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 Siapkan 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 [Menyiapkan AWS Config menggunakan Konsol](#) di Panduan AWS Config Pengembang.

Note

AWS Config merekam sumber daya Anda setelah menemukannya, yang mungkin memakan waktu beberapa menit.

Setelah AWS Config mulai merekam perubahan konfigurasi ke host dan instance Anda, Anda bisa mendapatkan riwayat konfigurasi host mana pun yang telah Anda alokasikan atau rilis dan instance apa pun yang telah Anda luncurkan, hentikan, atau hentikan. 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. Untuk informasi selengkapnya, lihat [Melihat Detail Konfigurasi Menggunakan CLI](#) di Panduan Developer AWS Config .
- Dengan menggunakan AWS Config API dalam aplikasi Anda. Pertama, Anda dapat menggunakan tindakan [ListDiscoveredResources](#) untuk mendapatkan daftar semua host dan instance. Kemudian, Anda dapat menggunakan [GetResourceConfigHistory](#) tindakan untuk mendapatkan detail konfigurasi host atau instance untuk interval waktu tertentu.

Misalnya, untuk mendapatkan daftar semua Host Khusus Anda 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 [Menyiapkan AWS Config 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 AWS Config Pengembang.
- API Amazon EC2: [GetResourceConfigHistory](#)

Instans Khusus

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 akun. AWS Ini berarti bahwa Instans Khusus secara fisik diisolasi pada tingkat perangkat keras host dari instans milik orang 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 memberikan visibilitas atau kontrol atas penempatan instans, dan instans tersebut tidak mendukung afinitas host. Jika Anda berhenti dan memulai Instans Khusus, 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 memberikan 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. 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 instans yang sepenuhnya didedikasikan untuk Anda gunakan.	Server fisik yang didedikasikan untuk satu akun pelanggan.
Pembagian kapasitas instans	Dapat berbagi kapasitas instans dengan akun lain.	Tidak didukung
Penagihan	Tagihan per host	Tagihan per instans
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
Pemulihan instans otomatis	Didukung. Untuk informasi selengkapnya, lihat Pemulihan host .	Didukung

	Host Khusus	Instans Khusus
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 [Host Khusus](#).

Topik

- [Dasar-dasar Instans Khusus](#)
- [Fitur yang didukung](#)
- [Batasan Instans Khusus](#)
- [Harga untuk Instans Khusus](#)
- [Bekerja dengan Instans Khusus](#)

Dasar-dasar Instans Khusus

VPC dapat memiliki penghunian default atau dedicated. Secara default, VPC Anda memiliki penghunian default dan instans yang diluncurkan ke VPC 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 [Buat VPC dengan penghunian instans khusus](#).
- Buat VPC dengan penghunian default dan tentukan secara manual penghunian dedicated untuk instans agar dijalankan sebagai Instans Khusus. Untuk informasi selengkapnya, lihat [Meluncurkan Instans Khusus ke dalam VPC](#).

Fitur yang didukung

Instans Khusus mendukung fitur dan integrasi AWS layanan berikut:

Topik

- [Instans Terpesan](#)
- [Penskalaan Otomatis](#)
- [Pemulihan otomatis](#)
- [Instans Spot Khusus](#)
- [Instance performa yang dapat melonjak](#)

Instans Terpesan

Untuk memesan kapasitas Instans Khusus, Anda dapat membeli Instans Cadangan Khusus atau Reservasi Kapasitas. Untuk informasi selengkapnya, lihat [Instans Terpesan](#) dan [Reservasi Kapasitas Sesuai Permintaan](#).

Saat Anda membeli Instans Terpesan Khusus, Anda membeli kapasitas untuk meluncurkan Instans Khusus ke dalam VPC dengan biaya penggunaan yang jauh lebih rendah; pengurangan harga dalam biaya penggunaan hanya berlaku jika Anda meluncurkan instans dengan penghunian 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 [Meluncurkan Instans Auto Scaling di VPC](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

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 memperbaiki. Untuk informasi selengkapnya, lihat [Ketahanan contoh](#).

Instans Spot Khusus

Anda dapat menjalankan Instans Spot Khusus dengan menentukan penghunian `dedicated` saat Anda membuat permintaan Instans Spot. Untuk informasi selengkapnya, lihat [Menentukan penghunian untuk Instans Spot Anda](#).

Instance performa yang dapat melonjak

Anda dapat memanfaatkan keuntungan menjalankan perangkat keras penghunian khusus dengan [the section called “Instans 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 kunci”](#).

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 fitur-fiturnya tidak didukung dengan VPC dengan penyewaan instance yang 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 dari harga untuk Instans Sesuai Permintaan. Untuk informasi selengkapnya, lihat [halaman produk Amazon EC2 Dedicated instans](#).

Bekerja dengan Instans Khusus

Anda dapat membuat VPC dengan tenancy penghunian instans dedicated untuk memastikan bahwa semua instans yang diluncurkan ke VPC adalah Instans Khusus. Atau, Anda dapat menentukan penghunian instans selama peluncuran.

Topik

- [Buat VPC dengan penghunian instans khusus](#)
- [Meluncurkan Instans Khusus ke dalam VPC](#)
- [Tampilkan informasi penghunian](#)
- [Mengubah penghunian suatu instans](#)
- [Mengubah penghunian VPC](#)

Buat VPC dengan penghunian instans khusus

Saat membuat VPC, Anda memiliki opsi untuk menentukan penghunian instansnya. Jika Anda meluncurkan instans ke dalam VPC yang memiliki penghunian instans dedicated, instans tersebut akan selalu berjalan sebagai Instans Khusus pada perangkat keras yang khusus untuk penggunaan Anda.

Untuk informasi selengkapnya tentang membuat VPC dan memilih opsi penghunian, lihat [Membuat VPC](#) di Panduan Pengguna Amazon VPC.

Meluncurkan Instans Khusus ke dalam VPC

Anda dapat meluncurkan Instans Khusus menggunakan wizard peluncuran instans Amazon EC2.

Console

Untuk meluncurkan Instans Khusus ke dalam VPC penghunian default menggunakan konsol

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.

 Note

Pastikan Anda memilih tipe instans yang didukung sebagai Instans Khusus. Untuk informasi selengkapnya, lihat [Amazon EC2 Dedicated instans](#).

5. Di bagian Key pair, pilih key pair untuk diasosiasikan dengan instans.
6. Di bagian Detail lanjutan, untuk Penghunian, pilih Khusus.
7. Konfigurasi opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
8. Pilih Luncurkan instans.

Command line

Untuk mengatur opsi penghunian untuk sebuah instans selama peluncuran menggunakan baris perintah

- [run-instances](#) (AWS CLI)
- [New-EC2Instance](#) (AWS Tools for Windows PowerShell)

Untuk informasi selengkapnya tentang meluncurkan instans dengan penghunian host, lihat [Luncurkan instans pada Host Khusus](#).

Tampilkan informasi penghunian

Console

Untuk menampilkan informasi penghunian untuk VPC Anda menggunakan konsol

1. Buka konsol Amazon VPC di <https://console.aws.amazon.com/vpc/>.
2. Pada panel navigasi, pilih VPC Anda.
3. Periksa penghunian instans di kolom Penghunian.
4. Jika kolom Tenancy tidak ditampilkan, pilih settings



di pojok kanan atas, aktifkan Tenancy, dan pilih Confirm.

Untuk menampilkan informasi penghunian untuk instans Anda menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Contoh.
3. Periksa penghunian instans di kolom Penghunian.
4. Jika kolom Tenancy tidak ditampilkan, lakukan salah satu hal berikut:
 - Pilih pengaturan  di pojok kanan atas, nyalakan Penyewaan, dan pilih Konfirmasi.
 - Pilih instans. Pada tab Detail yang ada dekat bagian bawah halaman, di bawah Host dan grup penempatan, periksa nilai untuk Penghunian.

Command line

Untuk mendeskripsikan penghunian VPC Anda menggunakan baris perintah

- [describe-vpcs](#) (AWS CLI)
- [Get-EC2Vpc](#) (AWS Tools for Windows PowerShell)

Untuk mendeskripsikan penghunian instans Anda menggunakan baris perintah

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell)

Untuk mendeskripsikan nilai penghunian dari Instans Terpesan menggunakan baris perintah

- [describe-reserved-instances](#) (AWS CLI)
- [Get-EC2ReservedInstance](#) (AWS Tools for Windows PowerShell)

Untuk mendeskripsikan nilai penghunian dari penawaran Instans Terpesan menggunakan baris perintah

- [describe-reserved-instances-offerings](#) (AWS CLI)
- [Get-EC2ReservedInstancesOffering](#) (AWS Tools for Windows PowerShell)

Mengubah penghunian suatu instans

Anda dapat mengubah penghunian instans yang dihentikan setelah peluncuran. Perubahan yang Anda buat berlaku saat berikutnya instans dimulai.

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. 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`.

Console

Untuk mengubah penghunian suatu instans menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pada panel navigasi, pilih Instans dan pilih instans Anda.
3. Pilih Status instans, Hentikan instan, Berhenti.
4. Pilih Tindakan, Pengaturan instans, Modifikasi penempatan instans.
5. Untuk Penghunian, pilih apakah akan menjalankan instans Anda pada perangkat keras khusus atau pada Host Khusus. Pilih Simpan.

Command line

Untuk mengubah nilai penghunian dari suatu instans menggunakan baris perintah

- [modify-instance-placement](#) (AWS CLI)
- [Edit-EC2InstancePlacement](#) (AWS Tools for Windows PowerShell)

Mengubah penghunian VPC

Anda dapat mengubah penghunian instans VPC dari `dedicated` menjadi `default` setelah Anda membuatnya. Mengubah penghunian instans dari VPC tidak akan memengaruhi penghunian instans yang ada di VPC tersebut. Saat berikutnya Anda meluncurkan sebuah instans di VPC, instans tersebut memiliki penghunian `default`, kecuali Anda menentukan sebaliknya selama peluncuran.

Note

Anda tidak dapat mengubah penghunian instans VPC dari `default` menjadi `dedicated` setelah dibuat.

Anda dapat memodifikasi penyewaan instans VPC hanya menggunakan AWS CLI, SDK, AWS atau Amazon EC2 API.

Command line

Untuk memodifikasi atribut penyewaan instance dari VPC menggunakan AWS CLI

Gunakan perintah [modify-vpc-tenancy](#) dan tentukan ID VPC serta nilai penghunian instans. Satu-satunya nilai yang di-support adalah `default`.

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

Reservasi Kapasitas

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

- Reservasi Kapasitas Sesuai Permintaan
- Blok Kapasitas untuk ML

Berikut adalah beberapa kasus penggunaan umum untuk Reservasi Kapasitas Sesuai Permintaan:

- Acara penskalaan — Buat Reservasi Kapasitas Sesuai Permintaan sebelum acara penting bisnis Anda untuk memastikan bahwa Anda dapat menskalakan saat diperlukan.
- Persyaratan peraturan dan pemulihan bencana — Gunakan Reservasi Kapasitas Sesuai Permintaan untuk memenuhi persyaratan peraturan untuk ketersediaan tinggi, dan pesan kapasitas di Zona Ketersediaan atau Wilayah yang berbeda untuk pemulihan bencana.

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 Kapasitas Sesuai Permintaan

Gunakan Reservasi Kapasitas Sesuai Permintaan jika Anda memiliki persyaratan kapasitas yang ketat, dan menjalankan beban kerja penting bisnis yang memerlukan jaminan kapasitas. Dengan Reservasi Kapasitas Sesuai Permintaan, 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.

Reservasi Kapasitas Sesuai Permintaan

Reservasi Kapasitas Sesuai Permintaan memungkinkan Anda untuk mencadangkan kapasitas komputasi untuk instans Amazon EC2 Anda di Zona Ketersediaan tertentu untuk durasi berapa pun. Reservasi Kapasitas mengurangi risiko tidak dapat memperoleh kapasitas Sesuai Permintaan jika ada kendala kapasitas. Jika Anda memiliki persyaratan kapasitas yang ketat, dan menjalankan beban kerja penting bisnis yang memerlukan 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 membutuhkannya.

Anda dapat membuat Reservasi Kapasitas kapan saja, tanpa membuat komitmen jangka waktu satu tahun atau tiga tahun. Kapasitas menjadi tersedia dan tagihan dimulai segera setelah Reservasi Kapasitas disediakan di akun Anda. Jika Anda tidak lagi membutuhkan jaminan kapasitas, batalkan Reservasi Kapasitas untuk melepaskan kapasitas agar tidak dikenai biaya. Anda juga dapat menggunakan diskon penagihan yang ditawarkan oleh Savings Plans dan Instans Terpesan Regional untuk mengurangi biaya Reservasi Kapasitas.

Saat Anda membuat Reservasi Kapasitas, Anda menentukan:

- Zona Ketersediaan tempat menyimpan kapasitas
- Jumlah instans untuk mencadangkan kapasitas
- Atribut instance, termasuk jenis instance, platform, Availability Zone, dan tenancy

Reservasi Kapasitas hanya dapat digunakan oleh instans yang cocok dengan atributnya. Secara default, mereka secara otomatis digunakan dengan menjalankan instans yang cocok dengan atribut. Jika Anda tidak memiliki instans yang sedang berjalan yang cocok dengan atribut Reservasi Kapasitas, instans tersebut tetap tidak digunakan sampai Anda meluncurkan sebuah instans dengan atribut yang cocok.

Daftar Isi

- [Perbedaan antara Reservasi Kapasitas, Instans Terpesan, dan Savings Plans](#)
- [Platform yang didukung](#)
- [Kuota](#)
- [Batasan](#)
- [Harga dan penagihan Reservasi Kapasitas](#)
- [Bekerja dengan Reservasi Kapasitas](#)
- [Bekerja dengan grup Reservasi Kapasitas](#)
- [Reservasi Kapasitas dalam grup penempatan kluster](#)
- [Reservasi Kapasitas di Local Zones](#)
- [Reservasi Kapasitas di Wavelength Zones](#)
- [Reservasi Kapasitas di AWS Outposts](#)
- [Bekerja dengan Reservasi Kapasitas bersama](#)

- [Armada Reservasi Kapasitas](#)
- [Memantau Reservasi Kapasitas](#)

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 diperlukan komitmen. Dapat dibuat dan dibatalkan sesuai kebutuhan.	Memerlukan komitmen tetap satu tahun atau tiga tahun		
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 Instans Sesuai Permintaan 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:

- [Instans Terpesan](#)
- [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 platform berikut:

- 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

Saat Anda membeli Reservasi Kapasitas, Anda harus menentukan platform yang mewakili sistem operasi untuk instans Anda.

- Untuk distribusi SUSE Linux dan RHEL, tidak termasuk BYOL, Anda harus memilih platform tertentu. Misalnya, platform SUSE Linux atau Red Hat Enterprise Linux.
- Untuk semua distribusi Linux lainnya (termasuk Ubuntu), pilih platform Linux/UNIX.
- Jika Anda membawa langganan RHEL (BYOL) yang sudah ada, maka Anda harus memilih platform Linux/UNIX.

- Untuk Windows dengan SQL Standard, Windows dengan SQL Server Enterprise, dan Windows dengan SQL Server Web, Anda harus memilih platform tertentu.
- Untuk semua versi Windows lainnya, kecuali BYOL yang tidak didukung, pilih platform Windows.

Kuota

Jumlah instans yang kapasitasnya dapat Anda pesan didasarkan pada kuota Instans Sesuai Permintaan akun Anda. Anda dapat memesan kapasitas untuk sebanyak mungkin instans sesuai kuota yang diizinkan, dikurangi jumlah instans yang sudah berjalan.

Kuota hanya berlaku untuk menjalankan instans. Jika instans Anda tertunda, berhenti, dihentikan, atau hibernasi, instans tersebut tidak akan diperhitungkan dalam kuota Anda.

Batasan

Sebelum Anda membuat Reservasi Kapasitas, perhatikan batasan dan larangan berikut.

- Reservasi Kapasitas yang aktif dan tidak terpakai diperhitungkan dalam batas Instans Sesuai Permintaan Anda.
- Reservasi Kapasitas tidak dapat dipindahtangankan dari satu AWS akun ke akun lainnya. Namun, Anda dapat berbagi Reservasi Kapasitas dengan AWS akun lain. Untuk informasi selengkapnya, lihat [Bekerja dengan 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).
- Reservasi Kapasitas tidak memastikan bahwa instans yang dihibernasi dapat melanjutkan setelah Anda mencoba untuk memulainya.

Harga dan penagihan Reservasi Kapasitas

Topik

- [Penetapan harga](#)
- [Penagihan](#)

- [Diskon tagihan](#)
- [Melihat tagihan Anda](#)

Penetapan harga

Reservasi Kapasitas ditagih dengan tarif Sesuai Permintaan yang setara, baik Anda menjalankan instans di kapasitas terpesan atau tidak. 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.

Reservasi Kapasitas ditagih dengan perincian per 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 Instans Linux `m4.large`, yang memiliki tarif Sesuai Permintaan USD0,10 per jam penggunaan. Dalam contoh ini, Reservasi Kapasitas disediakan di akun selama lima jam. Reservasi Kapasitas tidak digunakan untuk satu jam pertama, jadi akan ditagih untuk satu jam yang tidak digunakan dengan tarif Sesuai Permintaan standar tipe instans `m4.large`. 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 menempatnya. Pada jam keenam, Reservasi Kapasitas dibatalkan dan instans `m4.large` berjalan normal di luar kapasitas terpesan. Untuk jam tersebut, biaya dikenakan pada tarif Sesuai Permintaan untuk tipe instans `m4.large`.

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 Savings Plans dan Instans Cadangan Regional berlaku untuk Reservasi Kapasitas. AWS 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:

- [Instans Terpesan](#)
- [Panduan Pengguna Savings Plans](#)
- [Opsi tagihan dan pembelian](#)

Melihat tagihan Anda

Anda dapat meninjau biaya dan biaya ke akun Anda di AWS Billing and Cost Management konsol.

- 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 [Item Baris Reservasi Kapasitas](#) di Panduan Pengguna AWS Billing and Cost Management .

Bekerja dengan Reservasi Kapasitas

Untuk mulai menggunakan Reservasi Kapasitas, Anda membuat reservasi kapasitas di Zona Ketersediaan yang diperlukan. Kemudian, Anda dapat meluncurkan instans ke dalam kapasitas

terpesan, melihat pemanfaatan kapasitasnya dalam waktu nyata, dan menambah atau mengurangi kapasitasnya sesuai kebutuhan.

Secara default, Reservasi Kapasitas secara otomatis mencocokkan instance baru dan instance berjalan yang memiliki atribut yang cocok (jenis instans, platform, Availability Zone, dan tenancy). 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. Hal ini memungkinkan Anda untuk secara eksplisit mengontrol instans mana yang diizinkan untuk berjalan dalam kapasitas terpesan itu.

Anda dapat menentukan bagaimana reservasi berakhir. Anda dapat memilih untuk membatalkan Reservasi Kapasitas atau mengakhirinya secara otomatis pada waktu yang ditentukan. Jika Anda menentukan waktu berakhir, Reservasi Kapasitas dibatalkan dalam satu jam dari waktu yang ditentukan. Misalnya, jika Anda menentukan 31/5/2019, 13:30:55, Reservasi Kapasitas dijamin berakhir antara 13:30:55 dan 14:30:55 pada 31/5/2019. Setelah reservasi berakhir, 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 hingga Anda menghapus preferensi penargetan Reservasi Kapasitas atau mengonfigurasinya untuk menargetkan Reservasi Kapasitas yang berbeda.

Daftar Isi

- [Membuat Reservasi Kapasitas](#)
- [Luncurkan instans ke dalam Reservasi Kapasitas yang ada](#)
- [Memodifikasi Reservasi Kapasitas](#)
- [Untuk mengubah pengaturan Reservasi Kapasitas](#)
- [Melihat Reservasi Kapasitas](#)
- [Membatalkan Reservasi Kapasitas](#)

Membuat Reservasi Kapasitas

Jika permintaan Anda untuk membuat Reservasi Kapasitas berhasil, kapasitas akan segera tersedia. Kapasitas tetap dicadangkan untuk penggunaan Anda selama Reservasi Kapasitas aktif, dan Anda dapat meluncurkan instans ke dalamnya kapan saja. Jika Reservasi Kapasitas terbuka, instans baru dan instans yang ada yang memiliki atribut yang cocok secara otomatis berjalan dalam kapasitas Reservasi Kapasitas. Jika Reservasi Kapasitas targeted, instans harus secara khusus menargetkannya untuk dijalankan dalam kapasitas terpesan.

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 Instans Sesuai Permintaan Anda untuk keluarga instans yang dipilih. Tingkatkan batas Instans Sesuai Permintaan Anda untuk keluarga instans dan coba lagi. Untuk informasi selengkapnya, lihat [Kuota Instans Sesuai Permintaan](#).

Untuk membuat Reservasi Kapasitas menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pilih Reservasi Kapasitas, lalu pilih Buat Reservasi Kapasitas.
3. Di halaman Membuat Reservasi Kapasitas, konfigurasi pengaturan berikut di bagian Detail instans. Jenis instans, platform, Availability Zone, dan tenancy instance yang Anda luncurkan harus sesuai dengan jenis instans, platform, Availability Zone, dan tenancy yang Anda tentukan di sini atau Reservasi Kapasitas tidak diterapkan. Misalnya, jika Reservasi Kapasitas terbuka tidak cocok, peluncuran instans yang menargetkan Reservasi Kapasitas tersebut secara eksplisit akan gagal.
 - a. Tipe Instans—Tipe instans yang akan diluncurkan ke dalam kapasitas terpesan.
 - b. Luncurkan instans dengan pengoptimalan EBS—Tentukan apakah akan mencadangkan kapasitas untuk instans dengan pengoptimalan EBS. Opsi ini dipilih secara default untuk beberapa tipe instans. Untuk informasi selengkapnya, lihat [the section called “Optimisasi EBS”](#).
 - c. Platform—Sistem operasi untuk instans Anda. Untuk informasi selengkapnya, lihat [Platform yang didukung](#).
 - d. Zona Ketersediaan—Zona Ketersediaan tempat memesan kapasitas.
 - e. Penghunian—Tentukan apakah akan dijalankan pada perangkat keras bersama (default) atau instans khusus.
 - f. (Opsional) Grup penempatan ARN —ARN dari grup penempatan kluster tempat pembuatan Reservasi Kapasitas.

Untuk informasi selengkapnya, lihat [Reservasi Kapasitas dalam grup penempatan kluster](#).

Luncurkan instans ke dalam Reservasi Kapasitas yang ada

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. Anda hanya dapat meluncurkan instance ke Reservasi Kapasitas yang memiliki atribut yang cocok (jenis instans, platform, Availability Zone, dan tenancy) dan kapasitas yang memadai. 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.

Untuk meluncurkan instans ke dalam Reservasi Kapasitas yang ada menggunakan konsol

1. Ikuti prosedur untuk [meluncurkan instance](#), tetapi jangan meluncurkan instance sampai Anda menyelesaikan langkah-langkah berikut untuk menentukan pengaturan untuk grup penempatan dan Reservasi Kapasitas.
2. Perluas Detail lanjutan dan lakukan hal berikut:
 - a. Untuk grup Penempatan, pilih grup penempatan cluster untuk meluncurkan instance.
 - 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 kapasitas Sesuai Permintaan.
 - Buka — Meluncurkan instans 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 sesuai dengan kapasitas yang memadai, instans akan menggunakan kapasitas Sesuai Permintaan.
 - Target berdasarkan ID - 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.
 - Target menurut grup — Meluncurkan instans 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 sesuai dan kapasitas yang tersedia, instans diluncurkan ke dalam kapasitas Sesuai Permintaan.

3. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Untuk meluncurkan instance ke Reservasi Kapasitas yang ada menggunakan AWS CLI

Gunakan perintah [run-instances](#) dan tentukan parameter `--capacity-reservation-specification`.

Contoh berikut meluncurkan instans `t2.micro` ke dalam Reservasi Kapasitas terbuka apa pun yang memiliki atribut yang sesuai dan kapasitas yang tersedia:

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
--key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationPreference=open
```

Contoh berikut meluncurkan instans `t2.micro` ke dalam Reservasi Kapasitas targeted:

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
--key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

Contoh berikut meluncurkan instans `t2.micro` ke dalam grup Reservasi Kapasitas:

```
aws ec2 run-instances --image-id ami-abc12345 --count 1
--instance-type t2.micro --key-name MyKeyPair --subnet-
id subnet-1234567890abcdef1 --capacity-reservation-specification
CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-
groups:us-west-1:123456789012:group/my-cr-group}
```

Memodifikasi Reservasi Kapasitas

Anda dapat mengubah atribut Reservasi Kapasitas yang aktif setelah Anda membuatnya. Anda tidak dapat mengubah Reservasi Kapasitas setelah kedaluwarsa atau setelah Anda membatalkannya secara eksplisit.

Saat mengubah Reservasi Kapasitas, Anda hanya dapat menambah atau mengurangi kuantitas dan mengubah cara pelepasannya. Anda tidak dapat mengubah tipe instans, pengoptimalan EBS, platform, Zona Ketersediaan, atau kelayakan instans dari Reservasi Kapasitas. 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 menentukan jumlah baru yang melebihi batas Instans Sesuai Permintaan yang tersisa untuk tipe instans yang dipilih, pembaruan gagal.

Untuk mengubah Reservasi Kapasitas menggunakan konsol

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 Kuantitas atau Reservasi berakhir sesuai kebutuhan, dan pilih Simpan perubahan.

Untuk mengubah Reservasi Kapasitas menggunakan AWS CLI

Gunakan perintah [modify-capacity-reservation](#):

Misalnya, perintah berikut mengubah Reservasi Kapasitas untuk memesan kapasitas untuk delapan instans.

```
aws ec2 modify-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0 --  
instance-count 8
```

Untuk mengubah pengaturan Reservasi Kapasitas

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, Availability Zone, dan tenancy) 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.

Untuk mengubah pengaturan Reservasi Kapasitas sebuah instans menggunakan konsol

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 Tindakan, Pengaturan instans, Ubah Pengaturan Reservasi Kapasitas.

4. Untuk Reservasi Kapasitas, pilih salah satu opsi berikut:

- **Terbuka** — Meluncurkan instans ke Reservasi Kapasitas apa pun yang memiliki atribut yang sesuai dan kapasitas yang memadai untuk jumlah instans yang Anda pilih. Jika tidak ada Reservasi Kapasitas yang sesuai dengan kapasitas yang memadai, instans akan menggunakan kapasitas Sesuai Permintaan.
- **Tidak Ada** — Mencegah instans diluncurkan ke Reservasi Kapasitas. Instans berjalan dalam kapasitas Sesuai Permintaan.
- **Tentukan Reservasi Kapasitas** — Meluncurkan instans 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 instans ke dalam 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 sesuai dan kapasitas yang tersedia, instans diluncurkan ke dalam kapasitas Sesuai Permintaan.

Untuk mengubah setelan Reservasi Kapasitas instans menggunakan AWS CLI

Gunakan perintah [modify-instance-capacity-reservation-attributes](#).

Misalnya, perintah berikut mengubah pengaturan Reservasi Kapasitas instans menjadi open atau none.

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0 --capacity-reservation-specification CapacityReservationPreference=none | open
```

Misalnya, perintah berikut memodifikasi sebuah instans untuk menargetkan Reservasi Kapasitas tertentu.

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0 --capacity-reservation-specification CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

Misalnya, perintah berikut memodifikasi sebuah instans untuk menargetkan 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-1:123456789012:group/my-cr-group}
```

Melihat Reservasi Kapasitas

Reservasi Kapasitas memiliki kemungkinan status berikut:

- **active**—Kapasitas tersedia untuk digunakan.
- **expired**—Reservasi Kapasitas kedaluwarsa 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.

Note

Karena model [konsistensi akhirnya](#) yang diikuti oleh API Amazon EC2, setelah Anda membuat Reservasi Kapasitas, konsol dan respons [describe-capacity-reservations](#) dapat memerlukan waktu hingga 5 menit untuk menunjukkan bahwa Reservasi Kapasitas berada dalam status **active**. Selama waktu ini, konsol dan respons `describe-capacity-reservations` mungkin menunjukkan bahwa Reservasi Kapasitas dalam status **pending**. Namun, Reservasi Kapasitas mungkin sudah tersedia untuk digunakan dan Anda dapat mencoba meluncurkan instans ke dalamnya.

Untuk melihat Reservasi Kapasitas Anda menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pilih Reservasi Kapasitas dan pilih Reservasi Kapasitas untuk ditampilkan.
3. Pilih Lihat instans yang diluncurkan untuk reservasi ini.

Untuk melihat Reservasi Kapasitas Anda menggunakan AWS CLI

Gunakan perintah [describe-capacity-reservations](#):

Misalnya, perintah berikut menjelaskan semua Reservasi Kapasitas.

```
aws ec2 describe-capacity-reservations
```

Contoh keluaran

```
{
  "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"  
    }  
]  
}
```

Membatalkan Reservasi Kapasitas

Anda dapat membatalkan Reservasi Kapasitas kapan saja jika Anda tidak lagi membutuhkan kapasitas terpesan. Saat Anda membatalkan Reservasi Kapasitas, kapasitas segera dilepaskan dan tidak lagi dipesan untuk Anda gunakan.

Anda dapat membatalkan Reservasi Kapasitas yang kosong dan Reservasi Kapasitas yang memiliki instans berjalan. Jika Anda membatalkan Reservasi Kapasitas yang memiliki instans yang sedang berjalan, instans tersebut terus berjalan secara normal di luar reservasi kapasitas dengan tarif Instans Sesuai Permintaan standar atau dengan tarif diskon jika Anda memiliki Instans Savings Plans atau atau Instans Terpesan Regional.

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 [Untuk mengubah pengaturan Reservasi Kapasitas](#).

Untuk membatalkan Reservasi Kapasitas menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pilih Reservasi Kapasitas dan pilih Reservasi Kapasitas untuk dibatalkan.
3. Pilih Batalkan reservasi, Batalkan reservasi.

Untuk membatalkan Reservasi Kapasitas menggunakan AWS CLI

Gunakan perintah [cancel-capacity-reservation](#):

Misalnya, perintah berikut membatalkan Reservasi Kapasitas dengan ID `cr-1234567890abcdef0`.

```
aws ec2 cancel-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0
```

Bekerja dengan grup Reservasi Kapasitas

Anda dapat menggunakan AWS Resource Groups untuk membuat koleksi logis dari Reservasi Kapasitas, yang disebut grup sumber daya. Kelompok sumber daya adalah pengelompokan logis sumber AWS daya yang semuanya berada di AWS Wilayah yang sama. Untuk informasi selengkapnya tentang grup sumber daya, lihat [Apa Itu Grup Sumber Daya?](#) di Panduan Pengguna AWS Resource Groups .

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 sumber daya. Anda juga dapat menyertakan Reservasi Kapasitas yang memiliki atribut berbeda (jenis instans, platform, Availability Zone, dan tenancy) dalam satu grup sumber daya.

Saat Anda membuat grup sumber daya untuk Reservasi Kapasitas, Anda dapat menargetkan instans ke grup Reservasi Kapasitas alih-alih Reservasi Kapasitas individu. Instans yang menargetkan grup Reservasi Kapasitas cocok dengan Reservasi Kapasitas apa pun dalam grup yang memiliki atribut yang cocok (jenis instans, platform, Availability Zone, dan tenancy) dan kapasitas yang tersedia. Jika grup yang dipilih tidak memiliki Reservasi Kapasitas dengan atribut yang cocok dan kapasitas yang tersedia, instans berjalan menggunakan kapasitas Sesuai Permintaan. Jika Reservasi Kapasitas yang cocok ditambahkan ke grup yang ditargetkan di tahap selanjutnya, instans secara otomatis dicocokkan dengan dan dipindahkan ke kapasitas terpesan.

Untuk mencegah penggunaan Reservasi Kapasitas yang tidak disengaja dalam grup, konfigurasi Reservasi Kapasitas dalam grup untuk menerima hanya instans yang secara eksplisit menargetkan reservasi kapasitas. Untuk melakukan ini, atur kelayakan Instans menjadi tertarget (konsol lama) atau Hanya instans yang menentukan reservasi ini (konsol baru) saat membuat Reservasi Kapasitas menggunakan konsol Amazon EC2. Saat menggunakan AWS CLI, tentukan `--instance-match-criteria targeted` saat membuat reservasi kapasitas. Melakukan ini memastikan bahwa hanya instans yang secara eksplisit menargetkan grup, atau Reservasi Kapasitas dalam grup, yang dapat berjalan di grup.

Jika Reservasi Kapasitas dalam grup dibatalkan atau kedaluwarsa saat memiliki instans yang sedang berjalan, instans tersebut secara otomatis dipindahkan ke Reservasi Kapasitas lain dalam grup yang memiliki kecocokan atribut dan ketersediaan kapasitas. Jika tidak ada Reservasi Kapasitas yang tersisa di grup yang memiliki kecocokan atribut dan ketersediaan kapasitas, instans berjalan dalam kapasitas Sesuai Permintaan. Jika Reservasi Kapasitas yang cocok ditambahkan ke grup yang ditargetkan di tahap selanjutnya, instans secara otomatis dipindahkan ke kapasitas terpesan.

Topik

- [Membuat grup Reservasi Kapasitas](#)
- [Tambahkan Reservasi Kapasitas ke grup](#)
- [Melihat Reservasi Kapasitas dalam grup](#)
- [Tampilkan grup yang menjadi milik Reservasi Kapasitas](#)
- [Menghapus Reservasi Kapasitas dari grup](#)
- [Menghapus grup Reservasi Kapasitas](#)

Membuat grup Reservasi Kapasitas

Untuk membuat grup untuk Reservasi Kapasitas

Gunakan perintah [create-group](#) AWS CLI . Untuk name, berikan nama deskriptif untuk grup, dan untuk configuration, tentukan dua parameter permintaan Type:

- `AWS::EC2::CapacityReservationPool` untuk memastikan bahwa grup sumber daya dapat ditargetkan untuk peluncuran instans
- `AWS::ResourceGroups::Generic` dengan `allowed-resource-types` diatur ke `AWS::EC2::CapacityReservation` untuk memastikan bahwa grup sumber daya hanya menerima Reservasi Kapasitas

Misalnya, perintah berikut membuat grup bernama `MyCRGroup`.

```
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"]}]}'
```

Berikut ini adalah contoh output.

```
{
  "GroupConfiguration": {
    "Status": "UPDATE_COMPLETE",
    "Configuration": [
      {
        "Type": "AWS::EC2::CapacityReservationPool"
      },
      {
```

```

        "Type": "AWS::ResourceGroups::Generic",
        "Parameters": [
            {
                "Values": [
                    "AWS::EC2::CapacityReservation"
                ],
                "Name": "allowed-resource-types"
            }
        ]
    },
    "Group": {
        "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
        "Name": "MyCRGroup"
    }
}

```

Tambahkan Reservasi Kapasitas ke grup

Jika Anda menambahkan Reservasi Kapasitas yang dibagikan dengan Anda ke grup, dan Reservasi Kapasitas tersebut tidak dibagikan, tetapi akan dihapus secara otomatis dari grup.

Untuk menambahkan Reservasi Kapasitas ke grup

Gunakan perintah AWS CLI [group-resources](#). Untuk `group`, tentukan nama grup tempat Reservasi Kapasitas ditambahkan, dan untuk `resources`, tentukan ARN dari Reservasi Kapasitas yang akan ditambahkan. Untuk menambahkan banyak Reservasi Kapasitas, pisahkan ARN dengan spasi. Untuk menambahkan ARN Reservasi Kapasitas, gunakan [AWS CLI perintah deskripsi-kapasitas-reservasi dan tentukan ID Reservasi Kapasitas](#).

Misalnya, perintah berikut menambahkan dua Reservasi Kapasitas ke grup bernama MyCRGroup.

```

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

```

Berikut ini adalah contoh output.

```

{
  "Failed": [],

```

```
"Succeeded": [
  "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
  "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
]
```

Melihat Reservasi Kapasitas dalam grup

Untuk melihat Reservasi Kapasitas dalam grup tertentu

Gunakan perintah [list-group-resources](#) AWS CLI . Untuk group, tentukan nama grup.

Misalnya, perintah berikut menampilkan daftar Reservasi Kapasitas dalam grup bernama MyCRGroup.

```
aws resource-groups list-group-resources --group MyCRGroup
```

Berikut ini adalah contoh output.

```
{
  "QueryErrors": [],
  "ResourceIdentifiers": [
    {
      "ResourceType": "AWS::EC2::CapacityReservation",
      "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1"
    },
    {
      "ResourceType": "AWS::EC2::CapacityReservation",
      "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
    }
  ]
}
```

Note

Output perintah mencakup Reservasi Kapasitas yang Anda miliki dan Reservasi Kapasitas yang dibagikan dengan Anda.

Tampilkan grup yang menjadi milik Reservasi Kapasitas

AWS CLI

Untuk melihat grup tempat Reservasi Kapasitas tertentu ditambahkan

Gunakan perintah AWS CLI [get-groups-for-capacity-reservation](#).

Misalnya, perintah berikut menampilkan daftar grup tempat Reservasi Kapasitas `cr-1234567890abcdef1` ditambahkan.

```
aws ec2 get-groups-for-capacity-reservation --capacity-reservation-id cr-1234567890abcdef1
```

Berikut ini adalah contoh output.

```
{
  "CapacityReservationGroups": [
    {
      "OwnerId": "123456789012",
      "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup"
    }
  ]
}
```

Note

Jika Anda menentukan Reservasi Kapasitas yang dibagikan dengan Anda, perintah hanya menampilkan grup Reservasi Kapasitas yang Anda miliki.

Amazon EC2 console

Untuk melihat grup tempat Reservasi Kapasitas tertentu ditambahkan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Reservasi Kapasitas, pilih Reservasi Kapasitas untuk dilihat, lalu pilih Tampilkan.

Grup tempat Reservasi Kapasitas ditambahkan terdaftar di kartu Grup.

Note

Jika Anda memilih Reservasi Kapasitas yang dibagikan dengan Anda, konsol hanya menampilkan grup Reservasi Kapasitas yang Anda miliki.

Menghapus Reservasi Kapasitas dari grup

Untuk menghapus Reservasi Kapasitas dari grup

Gunakan perintah [AWS CLI ungroup-resources](#). Untuk `group`, tentukan ARN grup tempat menghapus Reservasi Kapasitas, dan untuk `resources`, tentukan ARN Reservasi Kapasitas yang akan dihapus. Untuk menghapus beberapa Reservasi Kapasitas, pisahkan ARN dengan spasi.

Contoh berikut menghapus dua Reservasi Kapasitas dari grup bernama `MyCRGroup`.

```
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
```

Berikut ini adalah contoh output.

```
{
  "Failed": [],
  "Succeeded": [
    "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd",
    "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
  ]
}
```

Menghapus grup Reservasi Kapasitas

Untuk menghapus grup

Gunakan perintah [hapus-grup](#) AWS CLI . Untuk `group`, berikan nama grup yang akan dihapus.

Misalnya, perintah berikut menghapus grup bernama `MyCRGroup`.

```
aws resource-groups delete-group --group MyCRGroup
```

Berikut ini adalah contoh output.

```
{
  "Group": {
    "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
    "Name": "MyCRGroup"
  }
}
```

Reservasi Kapasitas dalam grup penempatan klaster

Anda dapat membuat Reservasi Kapasitas dalam grup penempatan klaster untuk memesan kapasitas komputasi Amazon EC2 untuk beban kerja Anda. Grup penempatan klaster menawarkan manfaat latensi jaringan yang rendah dan throughput jaringan yang tinggi.

Membuat Reservasi Kapasitas di grup penempatan klaster memastikan bahwa Anda memiliki akses ke kapasitas komputasi di grup penempatan klaster saat Anda membutuhkannya, selama Anda membutuhkannya. Ini sangat ideal untuk memesan kapasitas untuk beban kerja berkinerja tinggi (HPC) yang memerlukan penskalaan komputasi. Hal ini memungkinkan Anda untuk menurunkan skala klaster Anda sambil memastikan bahwa kapasitas tetap tersedia untuk Anda gunakan sehingga Anda dapat meningkatkan skala kembali saat diperlukan.

Topik

- [Batasan](#)
- [Bekerja dengan Reservasi Kapasitas dalam grup penempatan klaster](#)

Batasan

Ingatlah hal-hal berikut saat membuat Reservasi Kapasitas dalam grup penempatan klaster:

- 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 tidak dapat membagikan Reservasi Kapasitas yang telah dibuat dalam grup penempatan klaster.
- Anda tidak dapat menghapus grup penempatan klaster yang memiliki Reservasi Kapasitas `active`. Anda harus membatalkan semua Reservasi Kapasitas di grup penempatan klaster sebelum Anda dapat menghapusnya.

Bekerja dengan Reservasi Kapasitas dalam grup penempatan klaster

Untuk mulai menggunakan Reservasi Kapasitas dengan grup penempatan klaster, lakukan langkah-langkah berikut.

Note

Jika Anda ingin membuat Reservasi Kapasitas di grup penempatan klaster yang ada, lewati Langkah 1. Kemudian, untuk Langkah 2 dan 3, tentukan ARN dari grup penempatan klaster yang ada. Untuk informasi tentang cara menemukan ARN dari grup penempatan klaster Anda yang ada, lihat. [Lihat informasi grup penempatan](#)

Topik

- [Langkah 1: \(Bersyarat\) Buat grup penempatan klaster untuk digunakan dengan Reservasi Kapasitas](#)
- [Langkah 2: Buat Reservasi Kapasitas di grup penempatan klaster](#)
- [Langkah 3: Luncurkan instans ke dalam grup penempatan klaster](#)

Langkah 1: (Bersyarat) Buat grup penempatan klaster untuk digunakan dengan Reservasi Kapasitas

Lakukan langkah ini hanya jika Anda perlu membuat grup penempatan klaster baru. Untuk menggunakan grup penempatan klaster yang ada, lewati langkah ini dan untuk Langkah 2 serta 3, gunakan ARN dari grup penempatan klaster tersebut. Untuk informasi tentang cara menemukan ARN dari grup penempatan klaster Anda yang ada, lihat. [Lihat informasi grup penempatan](#)

Anda dapat membuat grup penempatan klaster menggunakan salah satu metode berikut.

Console

Untuk membuat grup penempatan klaster menggunakan konsol

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 penempatan, pilih Klaster.
5. Pilih Buat grup.
6. Dalam tabel Grup penempatan, di kolom ARN Grup, catat ARN grup penempatan klaster yang Anda buat. Anda akan membutuhkannya untuk langkah selanjutnya.

AWS CLI

Untuk membuat grup penempatan cluster menggunakan AWS CLI

Gunakan perintah [create-placement-group](#). Untuk `--group-name`, tentukan nama deskriptif untuk grup penempatan, dan untuk `--strategy`, tentukan `cluster`.

Contoh berikut membuat grup penempatan bernama MyPG yang menggunakan strategi penempatan `cluster`.

```
aws ec2 create-placement-group \  
  --group-name MyPG \  
  --strategy cluster
```

Catat ARN grup penempatan yang ditampilkan dalam output perintah, karena Anda akan membutuhkannya untuk langkah berikutnya.

Langkah 2: Buat Reservasi Kapasitas di grup penempatan klaster

Anda membuat Reservasi Kapasitas dalam grup penempatan klaster dengan cara yang sama seperti Anda membuat Reservasi Kapasitas apa pun. Namun, Anda juga harus menentukan ARN grup penempatan klaster tempat membuat Reservasi Kapasitas. Untuk informasi selengkapnya, lihat [Membuat Reservasi Kapasitas](#).

Pertimbangan

- Grup penempatan klaster yang ditentukan harus dalam status `available`. Jika grup penempatan klaster berada dalam status `pending`, `deleting`, atau `deleted`, permintaan akan gagal.
- Reservasi Kapasitas dan grup penempatan klaster harus berada di Zona Ketersediaan yang sama. Jika permintaan untuk membuat Reservasi Kapasitas menentukan Zona Ketersediaan yang berbeda dari grup penempatan klaster, permintaan gagal.
- Anda dapat membuat Reservasi Kapasitas hanya untuk tipe instans yang didukung oleh grup penempatan klaster. Jika Anda menentukan tipe instans yang tidak didukung, permintaan gagal. Untuk informasi selengkapnya, lihat [Aturan dan batasan grup penempatan klaster](#).
- Jika Anda membuat Reservasi Kapasitas open dalam grup penempatan klaster dan ada instans berjalan yang memiliki atribut yang cocok (ARN grup penempatan, tipe instans, Zona Ketersediaan, platform, dan penghunian), 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 Instans Sesuai Permintaan Anda untuk keluarga instans yang dipilih. Tingkatkan batas Instans Sesuai Permintaan Anda untuk keluarga instans dan coba lagi. Untuk informasi selengkapnya, lihat [Kuota Instans Sesuai Permintaan](#).

Anda dapat membuat grup penempatan klaster menggunakan salah satu metode berikut.

Console

Untuk membuat Reservasi Kapasitas menggunakan konsol

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, Availability Zone, Tenancy, quantity, dan tanggal akhir sesuai kebutuhan.
4. Untuk grup Penempatan, pilih ARN dari grup penempatan klaster untuk membuat Reservasi Kapasitas.
5. Pilih Buat.

Untuk informasi selengkapnya, lihat [Membuat Reservasi Kapasitas](#).

AWS CLI

Untuk membuat Reservasi Kapasitas menggunakan AWS CLI

Gunakan perintah [create-capacity-reservation](#). Untuk `--placement-group-arn`, tentukan ARN grup penempatan klaster tempat membuat Reservasi Kapasitas.

```
$ aws ec2 create-capacity-reservation \  
  --instance-type instance_type \  
  --instance-platform platform \  
  --availability-zone az \  
  --instance-count quantity \  
  --placement-group-arn placement_group_ARN
```

Untuk informasi selengkapnya, lihat [Membuat Reservasi Kapasitas](#).

Langkah 3: Luncurkan instans ke dalam grup penempatan klaster

Anda meluncurkan instans ke Reservasi Kapasitas dalam grup penempatan klaster dengan cara yang sama seperti Anda meluncurkan instans ke Reservasi Kapasitas apa pun. Namun, Anda juga harus menentukan ARN grup penempatan klaster tempat peluncuran instans. Untuk informasi selengkapnya, lihat [Membuat Reservasi Kapasitas](#).

Pertimbangan

- Jika Reservasi Kapasitas adalah open, 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.
- Jika Reservasi Kapasitas hanya menerima peluncuran instans tertarget, Anda harus menentukan Reservasi Kapasitas target selain grup penempatan klaster dalam permintaan.
- Jika Reservasi Kapasitas hanya menerima peluncuran instans tertarget, Anda harus menentukan grup Reservasi Kapasitas target selain grup penempatan klaster dalam permintaan. Untuk informasi selengkapnya, lihat [Bekerja dengan grup Reservasi Kapasitas](#).

Anda dapat meluncurkan instans ke Reservasi Kapasitas di grup penempatan klaster menggunakan salah satu metode berikut.

Console

Untuk meluncurkan instans ke dalam Reservasi Kapasitas yang ada menggunakan konsol

1. Ikuti prosedur untuk [meluncurkan instance](#), tetapi jangan meluncurkan instance sampai Anda menyelesaikan langkah-langkah berikut untuk menentukan pengaturan untuk grup penempatan dan Reservasi Kapasitas.
2. Perluas Detail lanjutan dan lakukan hal berikut:
 - a. Untuk grup Penempatan, pilih grup penempatan cluster 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 klaster yang memiliki atribut yang cocok dan kapasitas yang memadai.
 - Target berdasarkan ID — Untuk meluncurkan instans ke Reservasi Kapasitas yang hanya menerima peluncuran instans yang ditargetkan.
 - Targetkan 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Untuk informasi selengkapnya, lihat [Luncurkan instans ke dalam Reservasi Kapasitas yang ada](#).

AWS CLI

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada menggunakan AWS CLI

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.

Perintah berikut meluncurkan instans ke Reservasi Kapasitas `targeted` dalam grup penempatan klaster.

```
$ aws ec2 run-instances \
```

```
--image-id ami_id \  
--count quantity \  
--instance-type instance_type \  
--key-name key_pair_name \  
--subnet-id subnetid \  
--capacity-reservation-specification  
CapacityReservationTarget={CapacityReservationId=capacity_reservation_id} \  
--placement "GroupName=cluster_placement_group_name"
```

Untuk informasi selengkapnya, lihat [Luncurkan instans ke dalam Reservasi Kapasitas yang ada](#).

Reservasi Kapasitas di Local Zones

Zona Lokal adalah perpanjangan dari AWS Wilayah 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 AWS Wilayah 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

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 [Menyertakan ke Local Zones](#).
2. Buat Reservasi Kapasitas di Zona Lokal. Untuk Zona Ketersediaan, pilih Zona Lokal. Zona Lokal diwakili oleh kode AWS Wilayah diikuti oleh pengidentifikasi yang menunjukkan lokasi, 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 Kapasitas Sesuai Permintaan, Anda dapat memilih Zona Wavelength dan Anda dapat meluncurkan instans ke dalam Reservasi Kapasitas dalam Zona Wavelength dengan menentukan subnet yang terkait dengan Zona Wavelength. Zona Wavelength diwakili oleh kode Wilayah AWS yang diikuti oleh pengidentifikasi yang menunjukkan lokasinya, misalnya us-east-1-w11-bos-w1z-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

Anda tidak dapat menggunakan grup Reservasi Kapasitas dalam Zona Wavelength.

Untuk menggunakan Reservasi Kapasitas di Zona Wavelength

1. Aktifkan Wavelength Zone untuk digunakan di akun Anda. AWS Untuk informasi selengkapnya, lihat [the section called “Aktifkan Zona Panjang Gelombang”](#).
2. Buat Reservasi Kapasitas di Zona Wavelength. Untuk Zona Ketersediaan, pilih Wavelength. Wavelength diwakili oleh kode Wilayah diikuti oleh AWS pengidentifikasi yang menunjukkan lokasi, misalnya. us-east-1-w11-bos-w1z-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](#).

Reservasi 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 AWS Wilayah, 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 suatu AWS Wilayah.

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 Outposts dengan akun AWS lain dalam organisasi Anda menggunakan AWS Resource Access Manager Untuk informasi selengkapnya tentang berbagi Reservasi Kapasitas, lihat [Bekerja dengan 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. Konfigurasi 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 .

Bekerja dengan Reservasi Kapasitas bersama

Pembagian Reservasi Kapasitas memungkinkan pemilik Reservasi Kapasitas untuk berbagi kapasitas cadangan mereka dengan AWS akun lain atau di dalam AWS organisasi. Hal ini memungkinkan Anda untuk membuat dan mengelola Reservasi Kapasitas secara terpusat, dan berbagi kapasitas cadangan di beberapa AWS akun atau dalam organisasi Anda AWS .

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 spesifik di dalam atau di luar AWS organisasinya
- Unit organisasi di dalam AWS organisasinya
- Seluruh AWS organisasinya

Daftar Isi

- [Prasyarat untuk berbagi Reservasi Kapasitas](#)
- [Layanan terkait](#)
- [Berbagi di seluruh Zona Ketersediaan](#)
- [Berbagi Reservasi Kapasitas](#)
- [Berhenti membagikan Reservasi Kapasitas](#)
- [Mengidentifikasi dan menampilkan Reservasi Kapasitas bersama](#)
- [Melihat penggunaan Reservasi Kapasitas bersama](#)
- [Izin Reservasi Kapasitas Bersama](#)
- [Tagihan dan pengukuran](#)
- [Batas instans](#)

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 atau AWS akun baru yang memiliki riwayat penagihan terbatas.
- Untuk berbagi Reservasi Kapasitas dengan AWS organisasi Anda 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 .

Layanan terkait

Pembagian Reservasi 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 berbagi sumber daya yang Anda miliki dengan membuat pembagian 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 selengkapnya AWS RAM, 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, Availability Zone us-east-1a untuk AWS akun Anda mungkin tidak memiliki lokasi yang sama dengan AWS akun lain. us-east-1a

Untuk mengidentifikasi lokasi Reservasi Kapasitas Anda yang terkait dengan akun Anda, Anda harus menggunakan ID Zona Ketersediaan (AZ ID). ID AZ adalah pengidentifikasi unik dan konsisten untuk Availability Zone di semua AWS akun. Misalnya, use1-az1 adalah ID AZ untuk us-east-1 Wilayah dan itu adalah lokasi yang sama di setiap AWS akun.

Untuk melihat ID AZ untuk Zona Ketersediaan di akun Anda

1. Buka AWS RAM konsol di <https://console.aws.amazon.com/ram>.
2. ID AZ untuk Wilayah saat ini ditampilkan di panel ID AZ Anda di sisi kanan layar.

Berbagi Reservasi Kapasitas

Saat Anda membagikan Reservasi Kapasitas yang Anda miliki dengan AWS akun lain, Anda mengaktifkannya untuk meluncurkan instans ke dalam 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, Availability Zone, dan tenancy) dan memiliki CapacityReservationPreference parameter yang disetel ke open, mereka 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 deskripsi-kapasitas-reservasi.](#) AWS CLI Jika Anda dapat melihat Reservasi Kapasitas bersama di konsol atau menjelaskannya menggunakan AWS CLI, itu tersedia untuk Anda gunakan dan Anda dapat meluncurkan instance ke dalamnya. Jika Anda mencoba meluncurkan instans ke dalam Reservasi Kapasitas dan instans tidak dapat diakses karena kegagalan berbagi, instans akan diluncurkan ke kapasitas Sesuai Permintaan.

Anda dapat membagikan Reservasi Kapasitas yang Anda miliki menggunakan konsol Amazon EC2, konsol AWS RAM , atau AWS CLI.

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 Sumber Daya Bersama](#) di Panduan Pengguna AWS RAM .

Untuk berbagi Reservasi Kapasitas yang Anda miliki menggunakan AWS CLI

Gunakan perintah [create-resource-share](#).

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 pembagian berhenti terus berjalan secara normal di luar kapasitas terpesan, dan kapasitas dikembalikan ke Reservasi Kapasitas sesuai dengan ketersediaan kapasitas Amazon EC2.
- 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. Anda dapat melakukan ini menggunakan konsol Amazon EC2, konsol AWS RAM , atau AWS CLI.

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 membagikan Reservasi Kapasitas yang Anda miliki menggunakan AWS RAM konsol

Lihat [Memperbarui Sumber Daya Bersama](#) di Panduan Pengguna AWS RAM .

Untuk berhenti membagikan Reservasi Kapasitas yang Anda miliki menggunakan AWS CLI

Gunakan perintah [disassociate-resource-share](#).

Mengidentifikasi dan menampilkan Reservasi Kapasitas bersama

 Important

Sebelum meluncurkan instans ke Reservasi Kapasitas yang dibagikan dengan Anda, pastikan bahwa Anda memiliki akses ke Reservasi Kapasitas bersama dengan melihatnya di konsol atau dengan menjelaskannya menggunakan perintah AWS CLI. Jika Anda dapat melihat Reservasi Kapasitas bersama di konsol atau menjelaskannya menggunakan AWS CLI, itu tersedia untuk Anda gunakan dan Anda dapat meluncurkan instance ke dalamnya. Jika Anda mencoba meluncurkan instans ke dalam Reservasi Kapasitas dan instans tidak dapat diakses karena kegagalan berbagi, instans akan diluncurkan ke kapasitas Sesuai Permintaan.

Pemilik dan konsumen dapat mengidentifikasi dan menampilkan Reservasi Kapasitas bersama menggunakan konsol Amazon EC2 dan AWS CLI.

Untuk mengidentifikasi Reservasi Kapasitas bersama menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Reservasi Kapasitas. Layar menampilkan daftar Reservasi Kapasitas yang Anda miliki dan Reservasi Kapasitas yang dibagikan dengan Anda. Kolom Pemilik menunjukkan ID AWS akun pemilik Reservasi Kapasitas. (me) di sebelah ID AWS akun menunjukkan bahwa Anda adalah pemiliknya.

Untuk mengidentifikasi Reservasi Kapasitas bersama menggunakan AWS CLI

Gunakan perintah [describe-capacity-reservations](#): Perintah mengembalikan Reservasi Kapasitas yang Anda miliki dan Reservasi Kapasitas yang dibagikan dengan Anda. OwnerId menunjukkan ID AWS akun pemilik Reservasi Kapasitas.

Melihat penggunaan Reservasi Kapasitas bersama

Pemilik Reservasi Kapasitas bersama dapat melihat penggunaannya kapan saja menggunakan konsol Amazon EC2 dan AWS CLI.

Untuk menampilkan penggunaan Reservasi Kapasitas 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 menampilkan penggunaan dan pilih tab Penggunaan.

Kolom ID akun AWS menunjukkan ID akun dari konsumen yang saat ini menggunakan Reservasi Kapasitas. Kolom Instans yang diluncurkan menunjukkan jumlah instans yang saat ini dijalankan setiap konsumen dalam kapasitas terpesan.

Untuk melihat penggunaan Reservasi Kapasitas menggunakan AWS CLI

Gunakan perintah [get-capacity-reservation-usage](#). AccountId menunjukkan ID akun dari akun yang menggunakan Reservasi Kapasitas. UsedInstanceCount menunjukkan jumlah instans yang saat ini dijalankan konsumen dalam kapasitas yang dicadangkan.

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.

Tagihan dan pengukuran

Tidak ada biaya tambahan untuk berbagi Reservasi Kapasitas.

Pemilik Reservasi Kapasitas ditagih untuk instans yang dijalankan di dalam Reservasi Kapasitas dan untuk kapasitas terpesan yang tidak digunakan. Konsumen ditagih untuk instans yang mereka jalankan di dalam Reservasi Kapasitas 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 Instans Sesuai Permintaan 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 dalam kapasitas bersama oleh konsumen diperhitungkan dalam batas Instans Sesuai Permintaan pemilik Reservasi Kapasitas. Batas instans Konsumen adalah jumlah dari batas Instans Sesuai Permintaan mereka sendiri dan kapasitas yang tersedia di Reservasi Kapasitas bersama yang aksesnya mereka miliki.

Armada Reservasi Kapasitas

Armada Reservasi Kapasitas Sesuai Permintaan adalah sekelompok 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](#)

- [Penetapan harga](#)
- [Konsep Armada Reservasi Kapasitas](#)
- [Bekerja dengan 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, instance baru dan instans yang sudah ada yang memiliki atribut yang cocok (tipe instans, platform, Availability Zone, dan tenancy) secara otomatis berjalan di Reservasi Kapasitas yang dibuat oleh Armada. Armada Reservasi Kapasitas tidak mendukung kriteria pencocokan instans target.

Pertimbangan

Ingatlah hal-hal berikut ini saat bekerja dengan Armada Reservasi Kapasitas:

- Armada Reservasi Kapasitas dapat dibuat, dimodifikasi, dilihat, dan dibatalkan menggunakan AWS API AWS CLI dan.
- 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 tag yang AWS dihasilkan 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 Armada Reservasi Kapasitas

Topik ini menjelaskan beberapa konsep Armada Reservasi Kapasitas.

Topik

- [Kapasitas target total](#)
- [Strategi alokasi](#)
- [Bobot tipe instans](#)
- [Prioritas tipe instans](#)

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 ($\text{total target capacity} / \text{instance type weight} = \text{number of instances}$).

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 kapasitas target total 384 unit menggunakan tipe dan bobot instans dalam contoh sebelumnya, Armada dapat memesan kapasitas untuk instans 6 `m4.16xlarge` ($384 \text{ kapasitas target total} / 64 \text{ bobot tipe instans} = 6 \text{ instans}$), 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.

Bekerja dengan Armada Reservasi Kapasitas

Topik

- [Sebelum Anda memulai](#)
- [Status Armada Reservasi Kapasitas](#)
- [Untuk mengubah Armada Reservasi Kapasitas](#)
- [Menampilkan Armada Reservasi Kapasitas](#)

- [Memodifikasi Armada Reservasi Kapasitas](#)
- [Membatalkan Armada Reservasi Kapasitas](#)

Sebelum Anda memulai

Sebelum Anda membuat Armada Reservasi Kapasitas:

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.

Status Armada Reservasi Kapasitas

Armada Reservasi Spot dapat berada dalam salah satu status berikut:

- `submitted` — Permintaan Armada Reservasi Kapasitas telah diajukan dan Amazon EC2 sedang bersiap untuk membuat Reservasi Kapasitas.
- `modifying` — Armada Reservasi Kapasitas sedang dimodifikasi. Armada tetap dalam status ini sampai modifikasi selesai.
- `active` — Armada Reservasi Kapasitas telah memenuhi kapasitas target totalnya dan berusaha mempertahankan kapasitas ini. Permintaan tetap berada dalam status ini sampai dimodifikasi atau dihapus.
- `partially_fulfilled` — Armada Reservasi Kapasitas telah memenuhi sebagian kapasitas target totalnya. Kapasitas Amazon EC2 tidak mencukupi untuk memenuhi total kapasitas target. Armada berusaha untuk secara asinkron memenuhi total kapasitas targetnya.
- `expiring` — Armada Reservasi Kapasitas telah mencapai tanggal berakhirnya dan sedang dalam proses kedaluwarsa. Satu atau beberapa Reservasi Kapasitasnya mungkin masih aktif.
- `expired` — Armada Reservasi Kapasitas telah mencapai tanggal berakhirnya. Armada dan Reservasi Kapasitasnya kedaluwarsa. Armada tidak dapat membuat Reservasi Kapasitas baru.

- `cancelling` — Armada Reservasi Kapasitas sedang dalam proses dibatalkan. Satu atau beberapa Reservasi Kapasitasnya mungkin masih aktif.
- `cancelled` — Armada Reservasi Kapasitas telah dibatalkan secara manual. Armada dan Reservasi Kapasitasnya dibatalkan dan Armada tidak dapat membuat Reservasi Kapasitas baru.
- `failed` — Armada Reservasi Kapasitas gagal untuk memesan kapasitas untuk tipe instans yang ditentukan.

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

Peran `AWSServiceRoleForEC2CapacityReservationFleet` terkait layanan dibuat secara otomatis di akun Anda saat pertama kali 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.

Anda dapat membuat Armada Reservasi Kapasitas hanya menggunakan baris perintah.

Untuk mengubah Armada Reservasi Kapasitas

Gunakan perintah [AWS CLI create-capacity-reservation-fleet](#).

```
aws ec2 create-capacity-reservation-fleet \  
--total-target-capacity capacity_units \  
--allocation-strategy prioritized \  
--instance-match-criteria open \  
--tenancy dedicated/default \  

```

```
--end-date yyyy-mm-ddThh:mm:ss.000Z \  
--instance-type-specifications file://instanceTypeSpecification.json
```

Berikut ini adalah isi dari `instanceTypeSpecification.json`.

```
[  
  {  
    "InstanceType": "instance_type",  
    "InstancePlatform": "platform",  
    "Weight": instance_type_weight,  
    "AvailabilityZone": "availability_zone",  
    "AvailabilityZoneId" : "az_id",  
    "EbsOptimized": true/false,  
    "Priority" : instance_type_priority  
  }  
]
```

Keluaran yang diharapkan

```
{  
  "Status": "status",  
  "TotalFulfilledCapacity": fulfilled_capacity,  
  "CapacityReservationFleetId": "cr_fleet_id",  
  "TotalTargetCapacity": capacity_units  
}
```

Contoh

```
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
```

`instanceTypeSpecification.json`

```
[  
  {  
    "InstanceType": "m5.xlarge",  
    "InstancePlatform": "Linux/UNIX",
```

```
"Weight": 3.0,  
"AvailabilityZone": "us-east-1a",  
"EbsOptimized": true,  
"Priority" : 1  
}  
]
```

Contoh keluaran

```
{  
  "Status": "submitted",  
  "TotalFulfilledCapacity": 0.0,  
  "CapacityReservationFleetId": "crf-abcdef01234567890",  
  "TotalTargetCapacity": 24  
}
```

Menampilkan Armada Reservasi Kapasitas

Anda dapat melihat informasi konfigurasi dan kapasitas untuk Armada Reservasi Kapasitas kapan saja. Menampilkan Armada juga memberikan detail tentang Reservasi Kapasitas individual yang ada di dalam Armada.

Anda dapat membuat Armada Reservasi Kapasitas hanya menggunakan baris perintah.

Untuk menampilkan Armada Reservasi Kapasitas

Gunakan [perintah AWS CLI deskripsi-kapasitas-reservasi-armada](#).

```
aws ec2 describe-capacity-reservation-fleets \  
--capacity-reservation-fleet-ids cr_fleet_ids
```

Output yang diharapkan

```
{  
  "CapacityReservationFleets": [  
    {  
      "Status": "status",  
      "EndDate": "yyyy-mm-ddThh:mm:ss.000Z",  
      "InstanceMatchCriteria": "open",  
      "Tags": [],  
      "CapacityReservationFleetId": "cr_fleet_id",  
      "Tenancy": "dedicated/default",  
      "InstanceTypeSpecifications": [  

```

```

    {
      "CapacityReservationId": "cr1_id",
      "AvailabilityZone": "cr1_availability_zone",
      "FulfilledCapacity": cr1_used_capacity,
      "Weight": cr1_instance_type_weight,
      "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
      "InstancePlatform": "cr1_platform",
      "TotalInstanceCount": cr1_number_of_instances,
      "Priority": cr1_instance_type_priority,
      "EbsOptimized": true/false,
      "InstanceType": "cr1_instance_type"
    },
  {
    "CapacityReservationId": "cr2_id",
    "AvailabilityZone": "cr2_availability_zone",
    "FulfilledCapacity": cr2_used_capacity,
    "Weight": cr2_instance_type_weight,
    "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
    "InstancePlatform": "cr2_platform",
    "TotalInstanceCount": cr2_number_of_instances,
    "Priority": cr2_instance_type_priority,
    "EbsOptimized": true/false,
    "InstanceType": "cr2_instance_type"
  },
],
"TotalTargetCapacity": total_target_capacity,
"TotalFulfilledCapacity": total_target_capacity,
"CreateTime": "yyyy-mm-ddThh:mm:ss.000Z",
"AllocationStrategy": "prioritized"
}
]
}

```

Contoh

```

aws ec2 describe-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890

```

Contoh Output

```

{
  "CapacityReservationFleets": [
    {

```

```

    "Status": "active",
    "EndDate": "2021-12-31T23:59:59.000Z",
    "InstanceMatchCriteria": "open",
    "Tags": [],
    "CapacityReservationFleetId": "crf-abcdef01234567890",
    "Tenancy": "default",
    "InstanceTypeSpecifications": [
      {
        "CapacityReservationId": "cr-1234567890abcdef0",
        "AvailabilityZone": "us-east-1a",
        "FulfilledCapacity": 5.0,
        "Weight": 1.0,
        "CreateDate": "2021-07-02T08:34:33.398Z",
        "InstancePlatform": "Linux/UNIX",
        "TotalInstanceCount": 5,
        "Priority": 1,
        "EbsOptimized": true,
        "InstanceType": "m5.xlarge"
      }
    ],
    "TotalTargetCapacity": 5,
    "TotalFulfilledCapacity": 5.0,
    "CreateTime": "2021-07-02T08:34:33.397Z",
    "AllocationStrategy": "prioritized"
  }
]
}

```

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.

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 dapat memodifikasi Armada Reservasi Kapasitas hanya menggunakan baris perintah.

Untuk mengubah Armada Reservasi Kapasitas

Gunakan perintah [modify-capacity-reservation-fleet](#) AWS CLI .

Note

Anda tidak dapat menentukan `--end-date` dan `--remove-end-date` dalam perintah yang sama.

```
aws ec2 modify-capacity-reservation-fleet \  
--capacity-reservation-fleet-id cr_fleet_ids \  
--total-target-capacity capacity_units \  
--end-date yyyy-mm-ddThh:mm:ss.000Z \  
--remove-end-date
```

Output yang diharapkan

```
{  
  "Return": true  
}
```

Contoh: Memodifikasi total kapasitas target

```
aws ec2 modify-capacity-reservation-fleet \  
--capacity-reservation-fleet-id crf-01234567890abcdef \  
--total-target-capacity 160
```

Contoh: Memodifikasi tanggal akhir

```
aws ec2 modify-capacity-reservation-fleet \  
--capacity-reservation-fleet-id crf-01234567890abcdef \  
--end-date 2021-07-04T23:59:59.000Z
```

Contoh: Menghapus tanggal akhir

```
aws ec2 modify-capacity-reservation-fleet \  
--capacity-reservation-fleet-id crf-01234567890abcdef \  
--remove-end-date
```

Contoh Output

```
{  
  "Return": true  
}
```

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 individual di Armada dibatalkan dan instans yang sebelumnya berjalan dalam kapasitas cadangan terus berjalan secara normal dalam kapasitas bersama.

Anda dapat membatalkan Armada Reservasi Kapasitas hanya menggunakan baris perintah.

Untuk membatalkan Armada Reservasi Kapasitas

Gunakan perintah [AWS CLI batal-kapasitas-reservasi-armada](#).

```
aws ec2 cancel-capacity-reservation-fleets \  
--capacity-reservation-fleet-ids cr_fleet_ids
```

Output yang diharapkan

```
{  
  "SuccessfulFleetCancellations": [  
    {  
      "CurrentFleetState": "state",  
      "PreviousFleetState": "state",  
      "CapacityReservationFleetId": "cr_fleet_id_1"  
    },  
    {  
      "CurrentFleetState": "state",  
      "PreviousFleetState": "state",  
      "CapacityReservationFleetId": "cr_fleet_id_2"  
    }  
  ]  
}
```

```

    ],
    "FailedFleetCancellations": [
      {
        "CapacityReservationFleetId": "cr_fleet_id_3",
        "CancelCapacityReservationFleetError": [
          {
            "Code": "code",
            "Message": "message"
          }
        ]
      }
    ]
  }
]
}

```

Contoh: Pembatalan yang berhasil

```

aws ec2 cancel-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890

```

Contoh Output

```

{
  "SuccessfulFleetCancellations": [
    {
      "CurrentFleetState": "cancelling",
      "PreviousFleetState": "active",
      "CapacityReservationFleetId": "crf-abcdef01234567890"
    }
  ],
  "FailedFleetCancellations": []
}

```

Contoh konfigurasi Armada Reservasi Kapasitas

Topik

- [Contoh 1: Kapasitas reservasi berdasarkan vCPU](#)

Contoh 1: Kapasitas reservasi berdasarkan vCPU

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 pada reservasi kapasitas untuk 30 instans `m5.4xlarge` ($480/16=30$), atau 10 instans `m5.12xlarge` ($480/48=10$).

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

Armada Reservasi Kapasitas Sesuai Permintaan menggunakan AWS Identity and Access Management peran terkait [layanan](#) (IAM). Peran tertaut layanan adalah tipe peran IAM unik yang tertaut langsung ke Armada Reservasi Kapasitas. Peran terkait layanan telah ditentukan sebelumnya oleh Armada Reservasi Kapasitas dan mencakup semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran tertaut layanan mempermudah pengaturan Armada Reservasi Kapasitas 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.

Izin peran tertaut layanan untuk Armada Reservasi Kapasitas

Armada Reservasi Kapasitas menggunakan peran terkait layanan yang diberi nama `AWSServiceRoleForEC2CapacityReservationFleet` untuk membuat, mendeskripsikan, memodifikasi, dan membatalkan Reservasi Kapasitas yang sebelumnya dibuat oleh Armada Reservasi Kapasitas, atas nama Anda.

Peran `AWSServiceRoleForEC2CapacityReservationFleet` terkait layanan mempercayai entitas berikut untuk mengambil peran: `capacity-reservation-fleet.amazonaws.com`

Peran menggunakan `AWSEC2CapacityReservationFleetRolePolicy` kebijakan, yang mencakup izin berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeCapacityReservations",
```

```

        "ec2:DescribeInstances"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateCapacityReservation",
        "ec2:CancelCapacityReservation",
        "ec2:ModifyCapacityReservation"
    ],
    "Resource": [
        "arn:aws:ec2:*:*:capacity-reservation/*"
    ],
    "Condition": {
        "StringLike": {
            "ec2:CapacityReservationFleet": "arn:aws:ec2:*:*:capacity-
reservation-fleet/crf-*"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags"
    ],
    "Resource": [
        "arn:aws:ec2:*:*:capacity-reservation/*"
    ],
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction": "CreateCapacityReservation"
        }
    }
}
]
}

```

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, silakan lihat [Izin Peran Tertaut Layanan](#) 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 akan dibuat secara otomatis 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 Anda membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat [Mengedit Peran Tertaut Layanan](#) dalam 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 `AWSServiceRoleForEC2CapacityReservationFleet` terkait layanan

1. Gunakan `delete-capacity-reservation-fleet` AWS CLI perintah atau `DeleteCapacityReservationFleet` API untuk menghapus Armada Reservasi Kapasitas di akun Anda.

- Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForEC2CapacityReservationFleet` terkait layanan. Untuk informasi selengkapnya, silakan lihat [Menghapus Peran Tertaut 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 Reservasi Kapasitas

Anda dapat menggunakan fitur berikut untuk memantau Reservasi Kapasitas Anda:

Topik

- [Pantau Reservasi Kapasitas menggunakan metrik CloudWatch](#)
- [Memantau Reservasi Kapasitas menggunakan EventBridge](#)
- [Notifikasi pemanfaatan](#)

Pantau Reservasi Kapasitas menggunakan metrik CloudWatch

Dengan CloudWatch metrik, Anda dapat memantau Reservasi Kapasitas secara efisien dan mengidentifikasi kapasitas yang tidak digunakan dengan menyetel CloudWatch 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 Sesuai Permintaan 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](#)
- [Lihat 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
<code>UsedInstanceCount</code>	Jumlah instans yang sedang digunakan. Unit: Jumlah
<code>AvailableInstanceCount</code>	Jumlah instans yang tersedia. Unit: Jumlah
<code>TotalInstanceCount</code>	Jumlah total instans yang telah Anda pesan. Unit: Jumlah
<code>InstanceUtilization</code>	Persentase instans kapasitas terpesan yang saat ini sedang digunakan. Satuan: Persen

Dimensi metrik Reservasi Kapasitas

Anda dapat menggunakan dimensi berikut untuk mempersempit metrik yang terdaftar pada tabel sebelumnya.

Dimensi	Deskripsi
<code>CapacityReservationId</code>	Dimensi unik global ini memfilter data yang Anda minta hanya untuk reservasi kapasitas yang teridentifikasi.

Lihat 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 konsol CloudWatch

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 dimensi metrik Berdasarkan Reservasi Kapasitas. Metrik akan dikelompokkan berdasarkan CapacityReservationId.
6. Untuk mengurutkan metrik, gunakan judul kolom. Untuk membuat grafik sebuah metrik, pilih kotak centang di sebelah metrik.

Untuk melihat metrik Reservasi Kapasitas (AWS CLI)

Gunakan perintah [list-metrics](#) berikut:

```
aws cloudwatch list-metrics --namespace "AWS/EC2CapacityReservations"
```

Memantau Reservasi Kapasitas menggunakan EventBridge

AWS Health mengirimkan acara ke Amazon EventBridge ketika Reservasi Kapasitas di akun Anda kurang dari 20 persen penggunaan selama periode tertentu. Dengan EventBridge, Anda dapat menetapkan aturan yang memicu tindakan terprogram dalam menanggapi 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 acara Amazon](#) di Panduan EventBridge Pengguna Amazon.

Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#).

Fitur ini tidak didukung di AWS GovCloud (US).

Daftar Isi

- [Peristiwa](#)
- [Buat EventBridge aturan](#)

Peristiwa

AWS Health mengirimkan peristiwa berikut ketika penggunaan kapasitas untuk Reservasi Kapasitas di bawah 20 persen.

Kejadian

- [AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION](#)
- [AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY](#)

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 penggunaan Reservasi Kapasitas turun di bawah 20 persen, buat topik Amazon SNS, lalu buat aturan untuk EventBridge `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.

Notifikasi pemanfaatan

AWS Health mengirimkan email dan AWS Health Dashboard 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 AWS Health Dashboard notifikasi 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 kirimkan acara ke EventBridge. Dengan EventBridge, Anda dapat membuat aturan yang mengaktifkan tindakan otomatis, seperti mengirim pemberitahuan email atau AWS Lambda fungsi pemicu, sebagai respons terhadap peristiwa tersebut. Untuk informasi selengkapnya, lihat [Memantau Reservasi Kapasitas menggunakan EventBridge](#).

Blok Kapasitas untuk ML

Blok Kapasitas untuk ML memungkinkan Anda untuk memesan instans GPU yang sangat dicari di masa mendatang untuk mendukung beban kerja machine learning (ML) berdurasi pendek. Instans yang berjalan di dalam Blok Kapasitas secara otomatis ditempatkan berdekatan di dalam [Amazon UltraClusters](#) EC2, untuk jaringan latensi rendah, skala petabit, dan tanpa pemblokiran.

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 sehari-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 saat ini tersedia untuk `p5.48xlarge` dan `p4d.24xlarge` contoh.

`p5.48xlarge` Instans tersedia di Wilayah AS Timur (Ohio) dan AS Timur (Virginia N.).

`p4d.24xlarge` Contoh tersedia di Wilayah AS Timur (Ohio) dan AS Barat (Oregon). Anda dapat memesan Blok Kapasitas dengan waktu mulai reservasi hingga delapan minggu ke depan.

Anda dapat menggunakan Blok Kapasitas untuk melakukan reservasi p5 dan p4d instans dengan opsi durasi reservasi dan kuantitas instans berikut.

- Durasi reservasi untuk kenaikan 1 hari hingga total 14 hari
- Opsi kuantitas instans reservasi dari 1, 2, 4, 8, 16, 32, atau 64 instans

Untuk memesan Blok Kapasitas, Anda mulai dengan menentukan kebutuhan kapasitas Anda, termasuk jenis instans, jumlah instans, 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 sebelum waktu Blok Kapasitas berakhir. Dengan 30 menit tersisa di reservasi Blok Kapasitas Anda, kami mulai menghentikan semua instans yang berjalan di Blok Kapasitas. Kami menggunakan waktu ini untuk membersihkan instans Anda sebelum mengirimkan Blok Kapasitas ke pelanggan berikutnya. 30 menit terakhir reservasi tidak dikenai biaya dalam harga Blok Kapasitas. Kami memancarkan acara melalui EventBridge 10 menit sebelum proses penghentian dimulai. Untuk informasi selengkapnya, lihat [Monitor Blok Kapasitas dengan EventBridge](#).

Topik

- [Platform yang didukung](#)
- [Pertimbangan](#)
- [Sumber daya terkait](#)
- [Harga dan penagihan Blok Kapasitas](#)
- [Bekerja dengan Blok Kapasitas](#)
- [Pantau Blok Kapasitas](#)

Platform yang didukung

Blok Kapasitas untuk ML saat ini mendukung p5.48xlarge dan p4d.24xlarge instance dengan penyewaan default. Saat Anda menggunakan AWS Management Console untuk membeli Blok Kapasitas, opsi platform default adalah Linux/UNIX. Saat Anda menggunakan AWS Command Line Interface (AWS CLI) atau AWS SDK untuk membeli Blok Kapasitas, opsi platform berikut tersedia:

- Linux/UNIX
- Linux Red Hat Enterprise
- RHEL dengan HA
- SUSE Linux
- Ubuntu Pro

Pertimbangan

Sebelum Anda menggunakan Blok Kapasitas, pertimbangkan detail dan batasan berikut.

- Blok Kapasitas dimulai dan diakhiri 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.
- Modifikasi dan pembatalan Blok Kapasitas tidak diizinkan.
- Blok Kapasitas tidak dapat dibagikan di seluruh AWS akun atau di dalam AWS Organisasi Anda.
- Blok Kapasitas tidak dapat digunakan dalam grup reservasi kapasitas.
- Jumlah total instans yang dapat dicadangkan di Blok Kapasitas di semua akun di AWS Organisasi Anda tidak dapat melebihi 64 instans pada tanggal tertentu.
- Untuk menggunakan Blok Kapasitas, instans harus secara khusus menargetkan ID reservasi.
- Instans dalam Blok Kapasitas tidak diperhitungkan dalam batas Instans Sesuai Permintaan Anda.
- Untuk instans P5 yang menggunakan AMI kustom, pastikan Anda memiliki [perangkat lunak dan konfigurasi yang diperlukan untuk EFA](#).
- Blok Kapasitas saat ini tidak dapat digunakan dengan grup node terkelola Amazon EKS atau Karpenter. Untuk informasi selengkapnya tentang cara membuat grup node yang dikelola sendiri Amazon EKS, lihat [Blok Kapasitas untuk ML](#) di Panduan Pengguna Amazon EKS.

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 ke Blok Kapasitas](#).
- Buat grup Auto Scaling Amazon EC2. Untuk informasi selengkapnya, lihat [Menggunakan Blok Kapasitas untuk beban kerja pembelajaran mesin](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Note

Jika Anda menggunakan Amazon EC2 Auto Scaling atau Amazon EKS, Anda dapat menjadwalkan penskalaan untuk dijalankan di awal reservasi Blok Kapasitas. Dengan penskalaan terjadwal, AWS secara otomatis menangani percobaan ulang untuk Anda, jadi Anda tidak perlu khawatir menerapkan logika coba lagi untuk menangani kegagalan sementara.

- 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 lebih lanjut tentang AWS ParallelCluster, lihat [Apa itu AWS ParallelCluster](#).

Harga dan penagihan Blok Kapasitas

Topik

- [Penetapan harga](#)
- [Penagihan](#)

Penetapan harga

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 suatu rentang tanggal, kami mengembalikan penawaran Blok Kapasitas dengan harga terendah yang tersedia. 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 akun AWS Anda dalam waktu 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 dalam waktu 12 jam, Blok Kapasitas Anda akan dilepas 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 Billing and Cost Management 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 [Penetapan harga](#). Tidak ada biaya tambahan untuk waktu yang tidak digunakan.

Untuk informasi selengkapnya, lihat [Melihat tagihan Anda](#) dalam Panduan Pengguna AWS Billing and Cost Management .

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. Dalam ID reservasi Blok Kapasitas Anda AWS Cost and Usage Report tercantum dalam item baris Reservasi/ReservationARN dari biaya dimuka Anda dan LinItem/ResourceID di tagihan ulang tahun Anda sehingga Anda dapat mengaitkan penggunaan dengan harga di muka yang sesuai.

Bekerja dengan Blok Kapasitas

Untuk mulai menggunakan Blok Kapasitas, pertama-tama temukan dan beli Blok Kapasitas yang tersedia yang sesuai dengan kebutuhan ukuran, durasi, dan waktu reservasi Anda. Kemudian, saat reservasi dimulai, Anda dapat menggunakan Blok Kapasitas dengan meluncurkan instans yang menargetkan ID reservasi. Tiga puluh menit sebelum reservasi berakhir, kami mulai mengakhiri semua instans yang masih berjalan di Blok Kapasitas.

Blok Kapasitas dikirimkan sebagai Reservasi Kapasitas `targeted` dalam satu Zona Ketersediaan. Untuk menjalankan instans di Blok Kapasitas, Anda harus menentukan ID reservasi saat meluncurkan instans Anda. Jika Anda menghentikan instans sendiri dan Blok Kapasitas kedaluwarsa, Anda tidak dapat memulai ulang hingga Anda menargetkan Blok Kapasitas lain dalam status `active`.

Secara default, Blok Kapasitas menghasilkan konektivitas jaringan dengan latensi rendah dan throughput tinggi di antara instans di dalam Blok Kapasitas, sehingga grup penempatan klaster dengan Blok Kapasitas tidak perlu digunakan.

Topik

- [Prasyarat](#)
- [Temukan dan beli Blok Kapasitas](#)
- [Luncurkan instans ke Blok Kapasitas](#)
- [Melihat Blok Kapasitas](#)

Prasyarat

Anda harus menggunakan yang sesuai Wilayah AWS untuk jenis instance yang ingin Anda gunakan. Untuk informasi selengkapnya, lihat [Wilayah](#).

Blok Kapasitas dengan `p5.48xlarge` instance tersedia sebagai berikut Wilayah AWS.

Nama Wilayah	Kode Wilayah
AS Timur (Ohio)	us-east-2
US East (N. Virginia)	us-east-1

Blok Kapasitas dengan p4d.24xlarge instans tersedia sebagai berikut Wilayah AWS.

Nama Wilayah	Kode Wilayah
AS Timur (Ohio)	us-east-2
AS Barat (Oregon)	us-west-2

 Note

Ukuran Blok Kapasitas 64 instans tidak didukung untuk semua jenis instans secara keseluruhan Wilayah AWS.

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 reservasi, tentukan.

- 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 memilih salah satu opsi berikut.

- Untuk durasi reservasi — Hingga 14 hari dengan kenaikan 1 hari
- Misalnya hitungan - 1, 2, 4, 8, 16, 32, atau 64 contoh

Jika Blok Kapasitas tersedia yang sesuai dengan spesifikasi Anda, kami akan mengembalikan detail penawaran Blok Kapasitas tunggal. Detail penawaran termasuk waktu mulai reservasi, Zona Ketersediaan untuk reservasi, dan harga reservasi. Untuk informasi selengkapnya, lihat [Penetapan harga](#).

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](#).

Anda dapat menggunakan salah satu metode berikut untuk menemukan dan membeli Blok Kapasitas.

Console

Untuk menemukan dan membeli Blok Kapasitas menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di bilah navigasi di bagian atas layar, pilih file 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, Beli Blok Kapasitas.
4. 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](#).
5. Di bawah Kapasitas total, pilih jumlah instans yang ingin Anda pesan.
6. Di bawah Durasi, masukkan jumlah hari yang Anda butuhkan untuk reservasi.
7. Pada Rentang tanggal untuk mencari Blok Kapasitas, masukkan tanggal mulai sedini mungkin dan tanggal akhir seakhir mungkin yang dapat diterima untuk reservasi Anda.
8. Pilih Temukan Blok Kapasitas.

9. Jika Blok Kapasitas tersedia yang memenuhi spesifikasi Anda, Anda akan melihat penawaran di bawah Blok Kapasitas yang Disarankan. Jika ada banyak penawaran yang memenuhi spesifikasi Anda, penawaran Blok Kapasitas dengan harga terendah yang tersedia akan ditampilkan. Untuk melihat penawaran Blok Kapasitas lainnya, sesuaikan input pencarian Anda dan pilih Temukan Blok Kapasitas lagi.
10. Ketika Anda menemukan penawaran Blok Kapasitas yang ingin Anda beli, pilih Berikutnya.
11. (Opsional) Pada halaman Tambahkan tanda, pilih Tambahkan tanda baru.
12. Halaman Tinjauan dan pembelian menampilkan daftar tanggal mulai dan berakhir, durasi, jumlah total instans, dan harga.

 Note

Blok Kapasitas tidak dapat dimodifikasi atau dibatalkan setelah Anda memesannya.

13. Di jendela popup Beli Blok Kapasitas, ketik konfirmasi, lalu pilih Beli.

AWS CLI

Untuk menemukan Blok Kapasitas menggunakan AWS CLI

Gunakan perintah `describe-capacity-block-offerings`.

Contoh berikut mencari Blok Kapasitas yang memiliki 16 instans `p5.48xlarge` dengan rentang tanggal mulai `2023-08-14` sampai `2023-10-22` dengan durasi 48 jam. Hitungan instans harus berupa bilangan bulat dari serangkaian opsi 1, 2, 4, 8, 16, 32, 64 yang telah ditentukan sebelumnya. Durasi kapasitas harus berupa bilangan bulat yang merupakan kelipatan 24 antara 24 dan 336, yang menunjukkan jumlah hari dalam jam.

```
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 48
```

Untuk membeli Blok Kapasitas menggunakan AWS CLI

Gunakan perintah `purchase-capacity-block` serta tentukan ID penawaran Blok Kapasitas yang ingin Anda beli dan platform instans.

```
aws ec2 purchase-capacity-block \  

```

```
--capacity-block-offering-id cbr-0123456789abcdefg \  
--instance-platform Linux/UNIX
```

Luncurkan instans ke 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`. Kami mengeluarkan acara melalui Amazon EventBridge untuk memberi tahu Anda bahwa Blok Kapasitas tersedia untuk digunakan. Untuk informasi selengkapnya, lihat [Pantau 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.

Langkah-langkah berikut menjelaskan cara meluncurkan instance ke dalam Blok Kapasitas di `active` negara bagian menggunakan AWS Management Console atau AWS CLI

Untuk informasi tentang cara mengatur grup simpul EKS agar secara otomatis menggunakan Blok Kapasitas saat dimulai, lihat [Blok Kapasitas untuk ML](#) di Panduan Pengguna Amazon EKS.

Untuk informasi tentang cara meluncurkan instans ke Blok Kapasitas menggunakan Armada EC2, lihat [Tutorial: Meluncurkan instans ke Blok Kapasitas](#).

Untuk informasi tentang cara membuat Templat peluncuran yang menargetkan Blok Kapasitas, lihat [Meluncurkan sebuah instans dari templat peluncuran](#).

Anda dapat menggunakan salah satu metode berikut untuk meluncurkan instans ke Blok Kapasitas.

Console

Untuk meluncurkan instans ke dalam Blok Kapasitas menggunakan konsol

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. (Opsional) Pada Nama dan tanda, Anda dapat memberi nama instans Anda dan menandai instans. Untuk informasi tentang tanda, lihat [Tandai sumber daya Amazon EC2 Anda](#)
5. Di bawah Gambar Aplikasi dan OS, pilih Amazon Machine Image (AMI).
6. Di bawah tipe instans, pilih tipe instans yang cocok dengan reservasi Blok Kapasitas Anda.
7. Di bawah 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](#).
8. Di bawah Pengaturan jaringan, gunakan pengaturan default, atau pilih Edit untuk mengonfigurasi pengaturan jaringan jika diperlukan.

Important

Instans Anda tidak dapat diluncurkan di subnet di Zona Ketersediaan yang berbeda dari Zona Ketersediaan tempat Blok Kapasitas Anda berada.

9. Di bawah Detail lanjutan, konfigurasi instans sebagai berikut.
 - a. Di bawah Opsi pembelian (tipe pasar), pilih Blok Kapasitas.
 - b. Pada Reservasi Kapasitas, pilih Target berdasarkan ID.
 - c. Pilih ID Reservasi Kapasitas dari reservasi Blok Kapasitas Anda.
10. Pada panel Ringkasan, untuk Jumlah instans, masukkan jumlah instans yang akan diluncurkan.
11. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instance ke Blok Kapasitas menggunakan AWS CLI

- Gunakan perintah `run-instances` dan tentukan `MarketType` dari `capacity-block` dalam struktur `instance-market-options`. Anda juga harus menentukan parameter `capacity-reservation-specification`.

Contoh berikut meluncurkan satu instans `p5.48xlarge` ke dalam Blok Kapasitas aktif yang memiliki kecocokan atribut dan ketersediaan kapasitas.

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 \  
  --instance-type p5.48xlarge --key-name MyKeyPair \  
  --subnet-id subnet-1234567890abcdef1 \  
  --instance-market-options MarketType='capacity-block' \  
  --capacity-reservation-specification \  
  CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

Melihat Blok Kapasitas

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.

Anda dapat menggunakan salah satu metode berikut untuk melihat reservasi Blok Kapasitas Anda.

Console

Untuk melihat Blok Kapasitas menggunakan konsol

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 menggunakan AWS CLI

Secara default, saat Anda menggunakan perintah [describe-capacity-reservations](#) baik Reservasi Kapasitas Sesuai Permintaan maupun reservasi Blok Kapasitas tercantum. Untuk melihat hanya reservasi Blok Kapasitas Anda, filter menggunakan `capacity-block` untuk parameter `capacity-reservation-type`.

Misalnya, perintah berikut menjelaskan satu atau beberapa reservasi Blok Kapasitas Anda saat ini Wilayah AWS.

```
aws ec2 describe-capacity-reservations --reservation-type capacity-block
```

Contoh keluaran

```
{
  "CapacityReservations": [
    {
      "CapacityReservationId": "cr-12345678",
```

```

    "EndDateType": "limited",
    "ReservationType": "capacity-block"
    "AvailabilityZone": "eu-east-2a",
    "InstanceMatchCriteria": "targeted",
    "EphemeralStorage": false,
    "CreateDate": "2023-11-29T14:22:45Z",
    "StartDate": "2023-12-15T12:00:00Z",
    "EndDate": "2023-08-19T12:00:00Z",
    "AvailableInstanceCount": 0,
    "InstancePlatform": "Linux/UNIX",
    "TotalInstanceCount": 16,
    "State": "payment-pending",
    "Tenancy": "default",
    "EbsOptimized": true,
    "InstanceType": "p5.48xlarge"
  },
  ...

```

Pantau Blok Kapasitas

Topik

- [Monitor Blok Kapasitas dengan EventBridge](#)
- [Kapasitas Pencatatan Memblokir panggilan API dengan AWS CloudTrail](#)

Monitor Blok Kapasitas dengan EventBridge

Saat reservasi Blok Kapasitas Anda dimulai, Amazon EC2 akan memancarkan peristiwa 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 kejadian yang berjalan dalam reservasi akan mulai berakhir dalam 10 menit. Untuk informasi selengkapnya tentang EventBridge acara, lihat [EventBridgeAcara Amazon](#).

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 Reservation 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 Reservation 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]"
  }
}

```

Kapasitas Pencatatan 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 mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan

yang dibuat untuk Blok Kapasitas, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

Informasi Blok Kapasitas di CloudTrail

CloudTrail diaktifkan pada 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 dalam riwayat Peristiwa. Anda dapat melihat, mencari, dan mengunduh acara terbaru di situs Anda Akun AWS. Untuk informasi selengkapnya, lihat [Melihat peristiwa dengan Riwayat CloudTrail acara](#).

Untuk catatan acara yang sedang berlangsung di Anda Akun AWS, termasuk acara untuk Blok Kapasitas, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan 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 AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- [Gambaran umum untuk membuat jejak](#)
- [CloudTrail layanan dan integrasi yang didukung](#)
- [Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail](#)
- [Menerima file CloudTrail log dari beberapa wilayah](#) dan [Menerima file CloudTrail log dari beberapa akun](#)

Semua tindakan Blok Kapasitas dicatat oleh CloudTrail dan didokumentasikan dalam Referensi API Amazon EC2. Misalnya, panggilan keCapacityBlockScheduled, dan CapacityBlockActive tindakan menghasilkan entri dalam file CloudTrail log.

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut ini:

- Apakah permintaan itu 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 terfederasi.

- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat elemen [CloudTrail UserIdentity](#).

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. Peristiwa mewakili permintaan tunggal dari sumber mana pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log untuk:

- [TerminateCapacityBlocksInstances](#)
- [CapacityBlockPaymentFailed](#)
- [CapacityBlockDijadwalkan](#)
- [CapacityBlockAktif](#)
- [CapacityBlockGagal](#)
- [CapacityBlockKedaluwarsa](#)

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 (N. Virginia):123456789012:instance/
i-1234567890abcdef0"
  }
  {
    "accountId": "123456789012",
    "type": "AWS::EC2::Instance",
    "ARN": "arn:aws::ec2:US East (N. Virginia):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 (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
    "accountId": "123456789012",
    "type": "AWS::EC2::CapacityReservation"
  }
],
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
  "capacityReservationId": "cr-12345678",
  "capacityReservationState": "payment-failed"
}
}

```

CapacityBlockDijadwalkan

```

{
  "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 (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ]
}

```

```
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
    "capacityReservationId": "cr-12345678",
    "capacityReservationState": "scheduled"
  }
}
```

CapacityBlockAktif

```
{
  "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 (N. Virginia):123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
    "capacityReservationId": "cr-12345678",
    "capacityReservationState": "active"
  }
}
```

CapacityBlockGagal

```
{
  "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 (N. Virginia):123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
    "capacityReservationId": "cr-12345678",
    "capacityReservationState": "failed"
  }
}
```

CapacityBlockKedaluwarsa

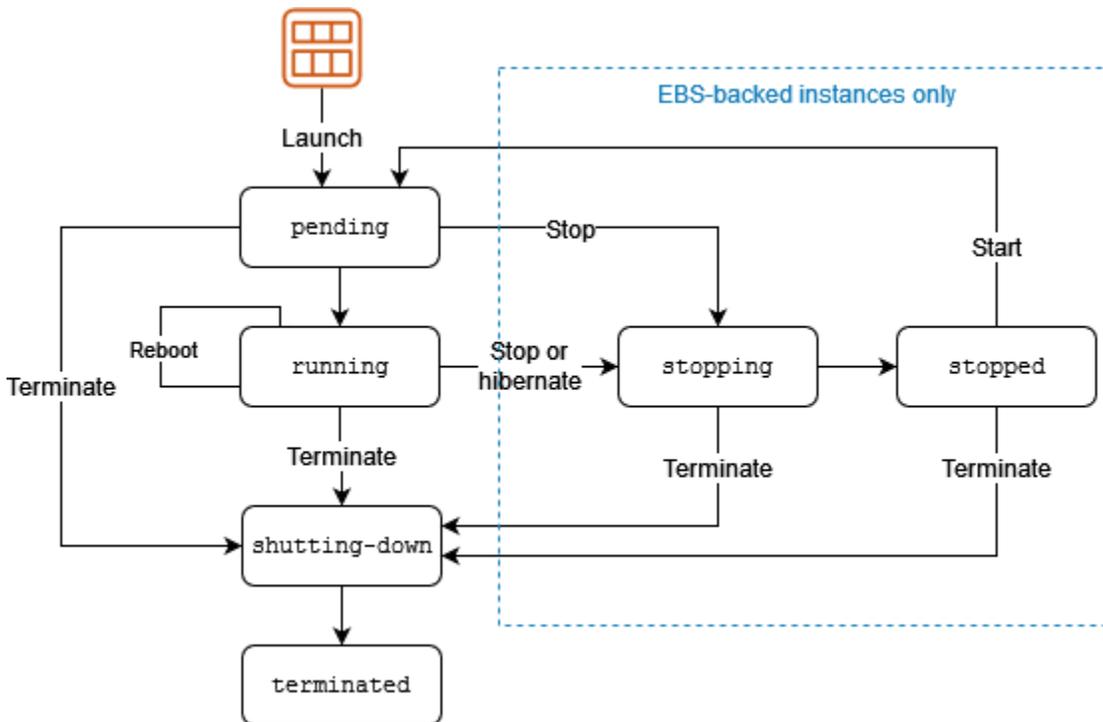
```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
```

```
"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 (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
    "accountId": "123456789012",
    "type": "AWS::EC2::CapacityReservation"
  }
],
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
  "capacityReservationId": "cr-12345678",
  "capacityReservationState": "expired"
}
}
```

Siklus hidup instans

Instans Amazon EC2 bertransisi melalui status yang berbeda dari saat Anda meluncurkannya hingga pengakhirannya.

Ilustrasi berikut menunjukkan transisi di antara status instans. Perhatikan bahwa Anda tidak dapat menghentikan dan memulai instans yang didukung penyimpanan instans. Untuk informasi selengkapnya tentang instans yang didukung penyimpanan instans, lihat [Penyimpanan untuk perangkat root](#).



Tabel berikut memberikan deskripsi singkat dari setiap status instance dan menunjukkan apakah penggunaan instance ditagih. Beberapa AWS sumber daya, seperti volume Amazon EBS dan alamat IP Elastis, 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
pending	Instans sedang bersiap untuk memasuki status running. Sebuah instans memasuki status pending saat diluncurkan atau ketika dimulai setelah berada dalam status stopped.	Tidak ditagih
running	Instans ini sedang berjalan dan siap digunakan.	Dikenakan biaya

Status instans	Deskripsi	Penagihan penggunaan instans
stopping	Instans sedang bersiap untuk dihentikan.	Tidak ditagih
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

 **Note**

Instans Terpesan yang diterapkan ke instans yang diakhiri akan ditagih hingga akhir jangka waktunya sesuai dengan opsi pembayaran mereka. Untuk informasi selengkapnya, lihat [Instans Terpesan](#)

Daftar Isi

- [Peluncuran instance](#)
- [Instans berhenti dan mulai \(hanya instans yang didukung Amazon EBS\)](#)
- [Hibernasi instans \(khusus instans yang didukung Amazon EBS\)](#)
- [Mulai ulang instans](#)
- [Pengakhiran instans](#)
- [Perbedaan antara boot ulang, penghentian, hibernasi, dan pengakhiran](#)
- [Luncurkan instans Anda](#)
- [Hentikan dan mulai instans Amazon EC2](#)
- [Hibernasi instans Amazon EC2 Anda](#)

- [Menyalakan ulang instans Anda](#)
- [Mengakhiri instans Amazon EC2](#)
- [Pensiun instans](#)
- [Ketahanan contoh](#)

Peluncuran instance

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. Kami menggunakan Amazon Machine Image (AMI) yang Anda tentukan saat peluncuran untuk booting instans. 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 berhenti dan mulai (hanya instans yang didukung Amazon EBS)

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, instans 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 [Hentikan dan mulai instans Amazon EC2](#).

Hibernasi instans (khusus instans yang didukung Amazon EBS)

Saat Anda menghibernasi sebuah instans, kami memberi sinyal pada sistem operasi untuk menjalankan hibernasi (`suspend-to-disk`), yang menyimpan konten dari memori instans (RAM) ke volume root Amazon EBS Anda. Kami mempertahankan volume root Amazon EBS instans dan semua volume data Amazon EBS yang terlampir. 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`. Kami tidak mengenakan biaya penggunaan untuk instans yang dihibernasi saat berada dalam status `stopped`, tetapi kami mengenakan biaya saat berada dalam status `stopping`, tidak seperti saat Anda [menghentikan instans](#) tanpa menghibernasinya. Kami tidak mengenakan biaya penggunaan untuk biaya transfer data, tetapi kami mengenakan biaya volume Amazon EBS, termasuk penyimpanan untuk data RAM.

Saat Anda memulai instans hibernasi, instans memasuki status `pending`, dan kami memindahkan instans ke komputer host baru (meskipun dalam beberapa kasus, instans tetap di 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](#).

Mulai ulang instans

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 [Menyalakan ulang instans Anda](#).

Pengakhiran instans

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 instans yang didukung Amazon EBS mendukung `InstanceInitiatedShutdownBehavior` atribut, yang mengontrol apakah instance berhenti atau berakhir saat Anda memulai shutdown dari dalam instance itu sendiri (misalnya, dengan menggunakan perintah di shutdown 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 perangkat root dan mempertahankan volume EBS lainnya.

Untuk informasi selengkapnya, lihat [Mengakhiri instans Amazon EC2](#).

Perbedaan antara boot ulang, penghentian, hibernasi, dan pengakhiran

Tabel berikut merangkum perbedaan utama antara me-reboot, menghentikan, hibernasi, dan menghentikan instans Anda.

Karakteristik	Mulai ulang	Hentikan/mulai (hanya instans yang didukung Amazon EBS)	Hibernasi (hanya instans yang didukung dengan Amazon EBS)	Mengakhiri
Komputer host	Instans tetap berada di komputer host yang sama	Kami memindahkan instans ke komputer host baru (meskipun dalam beberapa kasus, instans tetap di host saat ini).	Kami memindahkan instans ke komputer host baru (meskipun dalam beberapa kasus, instans tetap di host saat ini).	Tidak ada
Alamat IPv4 privat dan publik	Alamat ini tetap sama	Instans tetap mempertahankan alamat IPv4 privatnya. Instans tersebut ini mendapatkan alamat IPv4 publik baru, kecuali instans tersebut memiliki alamat IP Elastis, yang tidak berubah selama berhenti/dimulai.	Instans tetap mempertahankan alamat IPv4 privatnya. Instans tersebut ini mendapatkan alamat IPv4 publik baru, kecuali instans tersebut memiliki alamat IP Elastis, yang tidak berubah selama berhenti/dimulai.	Tidak ada
Alamat IP Elastis (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

Karakteristik	Mulai ulang	Hentikan/mulai (hanya instans yang didukung Amazon EBS)	Hibernasi (hanya instans yang didukung dengan Amazon EBS)	Mengakhiri
Alamat IPv6	Instans tetap mempertahankan alamat IPv6-nya	Instans tetap mempertahankan alamat IPv6-nya	Instans tetap mempertahankan alamat IPv6-nya	Tidak ada
Volume toko instan	Data disimpan	Datanya dihapus	Datanya dihapus	Datanya dihapus
Volume perangkat 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

Karakteristik	Mulai ulang	Hentikan/mulai (hanya instans yang didukung Amazon EBS)	Hibernasi (hanya instans yang didukung dengan Amazon EBS)	Mengakhiri
Penagihan	Jam penagihan instans tidak berubah	Anda tidak lagi dikenai biaya instans segera setelah statusnya berubah menjadi <code>stopping</code> . Setiap kali sebuah instans bertransisi dari <code>stopped</code> ke <code>running</code> , 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 <code>stopping</code> , tetapi tidak lagi dikenai biaya saat instans ada dalam status <code>stopped</code> . Setiap kali sebuah instans bertransisi dari <code>stopped</code> ke <code>running</code> , 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 <code>shutting-down</code>

Perintah pematian sistem operasi selalu mengakhiri instans yang didukung penyimpanan instans. Anda dapat mengontrol apakah perintah pematian sistem operasi akan menghentikan atau mengakhiri instans yang didukung Amazon EBS. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh instans](#).

Luncurkan instans Anda

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 mendaftar AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan Tingkat [AWS Gratis](#). Anda dapat menggunakan tingkat gratis untuk meluncurkan dan menggunakan

instans `t2.micro` secara gratis selama 12 bulan (di Wilayah yang tidak menyediakan `t2.micro`, Anda dapat menggunakan instans `t3.micro` pada tingkat gratis). Jika Anda meluncurkan instans yang tidak termasuk dalam tingkat gratis, Anda dikenai biaya penggunaan Amazon EC2 standar untuk instans tersebut. Untuk informasi selengkapnya, lihat [Harga Amazon EC2](#).

Anda dapat meluncurkan sebuah instans menggunakan metode berikut.

Metode	Dokumentasi
[Konsol Amazon EC2] Gunakan wizard peluncuran instans untuk menentukan parameter peluncuran.	Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama
[Konsol Amazon EC2] Buat templat peluncuran dan luncurkan instans dari templat peluncuran.	Meluncurkan sebuah instans dari templat peluncuran
[Konsol Amazon EC2] Gunakan instans yang ada sebagai basis.	Meluncurkan sebuah instans menggunakan parameter dari instans yang ada
[Konsol Amazon EC2] Gunakan AMI yang Anda beli dari AWS Marketplace.	Luncurkan sebuah AWS Marketplace instance
[AWS CLI] Gunakan AMI yang Anda pilih.	Menggunakan Amazon EC2 melalui AWS CLI
[AWS Tools for Windows PowerShell] Gunakan AMI yang Anda pilih.	Amazon EC2 dari AWS Tools for Windows PowerShell
[AWS CLI] Gunakan Armada EC2 untuk menyediakan kapasitas di berbagai tipe instans EC2 dan Availability Zone yang berbeda, dan di seluruh model pembelian Instans Sesuai Permintaan, Instans Cadangan, dan Instans Spot.	Armada EC2
[AWS CloudFormation] Gunakan AWS CloudFormation template untuk menentukan instance.	AWS::EC2::Instance di Panduan Pengguna AWS CloudFormation
[AWS SDK] Gunakan SDK khusus bahasa untuk AWS meluncurkan instance.	AWS SDK for .NET

Metode	Dokumentasi
	AWS SDK for C++
	AWS SDK for Go
	AWS SDK for Java
	AWS SDK untuk JavaScript
	AWS SDK for PHP V3
	AWS SDK untuk Python
	AWS SDK for Ruby V3

Note

Untuk meluncurkan instans EC2 ke subnet khusus IPv6, Anda harus menggunakan [Instans](#) yang dibangun di Sistem Nitro. AWS

Note

Saat meluncurkan instans khusus IPv6, ada kemungkinan bahwa DHCPv6 mungkin tidak segera menyediakan instans dengan server nama DNS IPv6. Selama penundaan awal ini, instans mungkin tidak dapat menyelesaikan domain publik.

Untuk instans yang berjalan di Amazon Linux 2, jika Anda ingin segera memperbarui file `/etc/resolv.conf` dengan server nama DNS IPv6, jalankan arahan cloud-init berikut saat peluncuran:

```
#cloud-config
bootcmd:
- /usr/bin/sed -i -E 's,^nameserver\s+[\.:digit:]]+$/,nameserver
fd00:ec2::253,' /etc/resolv.conf
```

Opsi lainnya adalah mengubah file konfigurasi dan menggambarkan ulang AMI Anda sehingga file tersebut langsung memiliki alamat server nama DNS IPv6 saat booting.

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 sebuah instans di Zona Lokal, Anda harus ikut serta dalam Zona Lokal, dan kemudian membuat subnet di zona tersebut. Untuk informasi selengkapnya, lihat [Memulai Local Zones](#)
- Zona Wavelength - Untuk meluncurkan instans di Zone Wavelength, Anda ikut serta dalam Zone Wavelength, lalu membuat subnet di zona tersebut. [Untuk informasi tentang cara meluncurkan instance di Wavelength Zone, lihat Memulai. AWS Wavelength](#)
- Outpost - Untuk meluncurkan instans di Outpost, Anda harus membuat Outpost. Untuk informasi tentang cara membuat Pos Luar, lihat [Memulai 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.

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.

Saat Anda selesai dengan sebuah instans, pastikan untuk mengakhirinya. Untuk informasi selengkapnya, lihat [Mengakhiri instans Amazon EC2](#).

Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru

Anda dapat meluncurkan sebuah instans menggunakan wizard peluncuran instans baru. Wizard peluncuran instans menentukan parameter peluncuran yang diperlukan untuk meluncurkan sebuah instans. Jika wizard peluncuran instans memberikan nilai default, Anda dapat menerima default atau menentukan nilai Anda sendiri. Jika Anda menerima nilai default, maka dimungkinkan untuk meluncurkan instans dengan memilih hanya key pair.

Important

Saat Anda meluncurkan sebuah instans yang tidak termasuk dalam [AWS Tingkat Gratis](#), Anda akan dikenai biaya untuk waktu instans tersebut berjalan, meskipun instans tetap idle.

Topik

- [Meluncurkan instans dengan cepat](#)
- [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#)
- [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#)

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. Untuk informasi tentang semua parameter dalam wizard peluncuran instans, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).

Untuk meluncurkan sebuah instans dengan cepat

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)). Pilih Wilayah tempat instans akan diluncurkan. Pilihan ini penting karena beberapa sumber daya Amazon EC2 dapat dibagikan di antara Wilayah, sedangkan sumber daya yang lainnya tidak. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
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

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. Parameter dikelompokkan dalam wizard peluncuran instans. Instruksi berikut membawa Anda melalui setiap kelompok parameter.

Parameter untuk konfigurasi instans

- [Memulai peluncuran instans](#)
- [Nama dan tanda](#)
- [Aplikasi dan Gambar OS \(Gambar Mesin Amazon\)](#)
- [Jenis instans](#)
- [Pasangan kunci \(login\)](#)
- [Pengaturan jaringan](#)
- [Mengonfigurasi penyimpanan](#)
- [Detail lanjutan](#)
- [Ringkasan](#)

Memulai peluncuran instans

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)). Pilih Wilayah tempat instans akan diluncurkan. Pilihan ini penting karena beberapa sumber daya Amazon EC2 dapat dibagikan di antara Wilayah, sedangkan sumber daya yang lainnya tidak. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.

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.

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 melalui semua AMI yang tersedia, masukkan kata kunci di bilah pencarian AMI dan kemudian tekan Enter. Untuk memilih AMI, pilih Pilih.

Terbaru

AMI yang baru saja Anda gunakan.

Pilih Baru diluncurkan atau Saat ini sedang digunakan, kemudian pilih AMI dari Amazon Machine Image (AMI).

AMI saya

AMI privat yang Anda miliki, atau AMI privat yang telah dibagikan dengan Anda.

Pilih Milik saya atau Dibagikan dengan saya, kemudian pilih AMI dari Amazon Machine Image (AMI).

Mulai Cepat

AMI 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 memenuhi syarat untuk tingkat gratis, pastikan bahwa AMI ditandai dengan Tingkat gratis yang memenuhi syarat.

Telusuri AMI lainnya

Pilih Telusuri AMI lainnya untuk menelusuri katalog lengkap AMI.

- Untuk menelusuri semua AMI yang tersedia, masukkan kata kunci di bilah pencarian 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 [Temukan AMI menggunakan parameter Systems Manager](#).
- Untuk mencari berdasarkan kategori, pilih AMI Mulai Cepat, AMI Saya, AMI AWS Marketplace, atau AMI Komunitas.

AWS Marketplace Ini adalah toko online tempat Anda dapat membeli perangkat lunak yang berjalan AWS, termasuk AMI. Untuk informasi selengkapnya tentang meluncurkan instance dari AWS Marketplace, lihat [Luncurkan sebuah AWS Marketplace instance](#). Di AMI Komunitas, Anda dapat menemukan AMI yang telah disediakan oleh anggota AWS untuk digunakan orang lain. AMI dari Amazon atau mitra terverifikasi ditandai sebagai Penyedia terverifikasi.

- Untuk memfilter daftar AMI, 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 mana tipe AMI yang Anda butuhkan: baik ebs (didukung oleh Amazon EBS) atau instance-store (didukung oleh penyimpanan instans. Untuk informasi selengkapnya, lihat [Penyimpanan untuk perangkat root](#).
- Periksa tipe Virtualisasi yang terdaftar untuk setiap AMI. Perhatikan mana tipe AMI yang Anda butuhkan, baik hvm atau paravirtual. Sebagai contoh, beberapa tipe instans memerlukan HVM. Untuk informasi selengkapnya tentang jenis virtualisasi Linux, lihat [Tipe virtualisasi AMI](#).
- Periksa Mode booting yang terdaftar untuk setiap AMI. Perhatikan mana AMI yang menggunakan mode booting yang Anda butuhkan: baik legacy-bios, uefi, atau uefi-preferred. Untuk informasi selengkapnya, lihat [Mode boot Amazon EC2](#).
- Pilih AMI yang memenuhi kebutuhan Anda, lalu pilih Pilih.

Peringatan saat mengganti AMI

Jika Anda mengubah konfigurasi volume atau grup keamanan apa pun yang terkait dengan AMI yang dipilih, kemudian Anda memilih AMI yang berbeda, sebuah jendela akan 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.

Jenis 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 instans Amazon EC2](#).

- Untuk Tipe instans, pilih tipe instans untuk instans tersebut.

Tingkat Gratis - Jika AWS akun Anda berusia 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). Jika tipe instans memenuhi syarat untuk masuk Tingkat Gratis, instans tersebut diberi label Memenuhi syarat Tingkat Gratis. Untuk informasi selengkapnya tentang t2.micro dan t3.micro, lihat [Instans performa yang dapat melonjak](#).

- Bandingkan tipe instans: Anda dapat membandingkan tipe instans yang berbeda dengan atribut berikut: jumlah vCPU, arsitektur, jumlah memori (GiB), jumlah penyimpanan (GB), tipe penyimpanan, dan performa jaringan.
- Dapatkan saran: Anda bisa mendapatkan panduan dan saran mengenai tipe instans dari pemilih tipe instans Amazon Q EC2. Untuk informasi selengkapnya, lihat [Dapatkan rekomendasi tipe instans untuk beban kerja baru](#).

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

Konfigurasi pengaturan jaringan, sesuai keperluan.

- VPC: Pilih VPC yang ada untuk instans Anda. Anda dapat memilih VPC default atau VPC yang Anda buat. Untuk informasi selengkapnya, lihat [the section called “Virtual private cloud”](#).

- Subnet: Anda dapat meluncurkan sebuah instans di subnet yang terkait dengan Zona Ketersediaan, Local Zone, Wavelength Zone, atau Outpost.

Untuk meluncurkan instans di Zona Ketersediaan, pilih subnet tempat Anda akan meluncurkan instans. Untuk membuat subnet baru, pilih Buat subnet baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke wizard peluncuran instans dan pilih ikon Segarkan untuk memuat subnet Anda dalam daftar.

Untuk meluncurkan instans di subnet khusus IPv6, instans harus [dibangun di Sistem Nitro](#).

Untuk meluncurkan instans di Local Zone, pilih subnet yang Anda buat di Local Zone.

Untuk meluncurkan sebuah instans di Outpost, pilih subnet di VPC yang Anda kaitkan dengan Outpost.

- Tetapkan otomatis IP Publik: Tentukan apakah instans Anda menerima alamat IPv4 publik yang lain. Secara default, instans di subnet default menerima alamat IPv4 publik, sedangkan instans di subnet nondefault tidak menerimanya. Anda dapat memilih Aktifkan atau Nonaktifkan untuk mengganti pengaturan default subnet. Untuk informasi selengkapnya, lihat [Alamat IPv4 publik](#).
- Firewall (grup keamanan): Gunakan grup keamanan untuk menentukan aturan firewall bagi instans Anda. Aturan ini menentukan lalu lintas jaringan yang masuk yang dikirim ke instans Anda. Semua lalu lintas lainnya diabaikan. Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).

Jika Anda menambahkan antarmuka jaringan, Anda harus menentukan grup keamanan yang sama di antarmuka jaringan.

Pilih atau buat grup keamanan sebagai berikut:

- Untuk memilih grup keamanan yang ada untuk VPC Anda, pilih Pilih grup keamanan yang ada, dan pilih grup keamanan Anda dari Grup keamanan umum.
- Untuk membuat grup keamanan baru untuk VPC Anda, pilih Buat grup keamanan. Wizard peluncuran instans secara otomatis menentukan grup keamanan launch-wizard-x dan menyediakan kotak centang berikut untuk menambahkan aturan grup keamanan dengan cepat:

(Linux) Izinkan lalu lintas SSH dari - Membuat aturan masuk untuk memungkinkan Anda terhubung ke instance Anda melalui SSH (port 22).

(Windows) Izinkan lalu lintas RDP dari - Membuat aturan masuk untuk memungkinkan Anda terhubung ke instans Anda melalui RDP (port 3389).

Tentukan apakah lalu lintas berasal dari Mana saja, Kustom, atau IP Saya.

Izinkan lalu lintas HTTP dari internet – Membuat aturan masuk yang membuka port 443 (HTTP) untuk memungkinkan lalu lintas internet dari mana saja. Jika instans Anda akan menjadi server web, Anda akan memerlukan aturan ini.

Izinkan lalu lintas HTTP dari internet – Membuat aturan masuk yang membuka port 80 (HTTP) untuk memungkinkan lalu lintas internet dari mana saja. Jika instans Anda akan menjadi server web, Anda akan memerlukan aturan ini.

Anda dapat mengedit aturan ini dan menambahkan aturan untuk menyesuaikan dengan kebutuhan Anda.

Untuk mengedit atau menambahkan aturan, pilih Edit (di kanan atas). Untuk menambahkan aturan, pilih Tambahkan aturan grup keamanan. Untuk Tipe, pilih tipe lalu lintas jaringan. Bidang Protokol secara otomatis diisi dengan protokol untuk membuka lalu lintas jaringan. Untuk Tipe sumber, pilih tipe sumber. Untuk mengizinkan wizard peluncuran instans menambahkan alamat IP publik komputer Anda, pilih IP Saya. Jika Anda terhubung melalui ISP atau dari belakang firewall Anda tanpa alamat IP statis, maka Anda harus menemukan rentang alamat IP yang digunakan oleh komputer klien.

Warning

Aturan yang mengaktifkan semua alamat IP ($0.0.0.0/0$) untuk mengakses instans Anda melalui SSH atau RDP dapat diterima jika Anda meluncurkan instans pengujian sebentar dan akan menghentikan atau mengakhirinya segera, tetapi tidak aman untuk lingkungan produksi. Anda hanya boleh mengotorisasi alamat IP atau rentang alamat tertentu saja untuk mengakses instans.

- Konfigurasi jaringan lanjutan – Hanya tersedia jika Anda memilih subnet.

Antarmuka jaringan

- Indeks perangkat: Indeks kartu jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0. Beberapa tipe instans mendukung banyak kartu jaringan.
- Antarmuka jaringan: Pilih Antarmuka baru agar Amazon EC2 dapat membuat antarmuka baru, atau memilih antarmuka jaringan yang ada dan tersedia.
- Deskripsi: (Opsional) Deskripsi untuk antarmuka jaringan baru.

- **Subnet:** Subnet tempat membuat antarmuka jaringan baru. Untuk antarmuka jaringan primer (eth0), ini adalah subnet tempat instans diluncurkan. Jika Anda telah memasukkan antarmuka jaringan yang ada untuk eth0, instans akan diluncurkan di subnet tempat antarmuka jaringan berada.
- **Grup keamanan:** Satu atau beberapa grup keamanan di VPC Anda yang akan digunakan untuk mengaitkan antarmuka jaringan.
- **IP Primer:** Alamat IPv4 privat dari jangkauan rentang subnet Anda. Biarkan kosong agar Amazon EC2 dapat memilih alamat IPv4 privat untuk Anda.
- **IP Sekunder:** Satu atau beberapa alamat IPv4 privat tambahan dari jangkauan subnet Anda. Pilih **Tetapkan secara manual** dan masukkan alamat IP. Pilih **Tambahkan IP** untuk menambahkan alamat IP lain. Atau, pilih **Tetapkan secara otomatis** agar Amazon EC2 dapat memilih salah satu untuk Anda, dan masukkan nilai yang menunjukkan jumlah alamat IP yang akan ditambahkan.
- **(IPv6 saja) IP IPv6:** Alamat IPv6 dari rentang subnet. Pilih **Tetapkan secara manual** dan masukkan alamat IP. Pilih **Tambahkan IP** untuk menambahkan alamat IP lain. Atau, pilih **Tetapkan secara otomatis** agar Amazon EC2 dapat memilih salah satu untuk Anda, dan masukkan nilai yang menunjukkan jumlah alamat IP yang akan ditambahkan.
- **Prefiks IPv4:** Awalan IPv4 untuk antarmuka jaringan.
- **Prefiks IPv6:** Awalan IPv6 untuk antarmuka jaringan.
- **(Tumpukan ganda dan IPv6 saja) Menetapkan IP IPv6 Primer:** (Opsional) Jika Anda meluncurkan instans ke subnet tumpukan ganda atau IPv6 saja, Anda memiliki opsi untuk Menetapkan IP IPv6 Primer. Dengan menetapkan alamat IPv6 primer, Anda akan dapat menghindari mengganggu lalu lintas ke instans atau ENI. Pilih **Aktifkan** jika instans ini bergantung pada alamat IPv6 yang tidak berubah. Saat Anda meluncurkan instance, secara otomatis AWS akan menetapkan alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instans Anda menjadi alamat IPv6 utama. Setelah Anda mengaktifkan alamat IPv6 GUA 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 akan menjadi alamat IPv6 utama primer.
- **Hapus saat pengakhiran:** Apakah antarmuka jaringan akan dihapus saat instans dihapus.
- **Elastic Fabric Adapter:** Menunjukkan apakah antarmuka jaringan adalah Elastic Fabric Adapter. Untuk informasi selengkapnya, lihat [Elastic Fabric Adapter](#).

- **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 peluncuran instans tidak menyertakan konfigurasi ENA Ekspres untuk instans 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 peluncuran instans tidak menyertakan konfigurasi ENA Ekspres untuk instans kecuali Anda memilih Aktifkan atau Nonaktifkan.

Pilih Tambahkan antarmuka jaringan untuk menambahkan antarmuka jaringan tambahan. Antarmuka jaringan tambahan dapat berada di subnet yang berbeda dari VPC yang sama atau di subnet di VPC berbeda yang Anda miliki (selama subnet berada di Availability Zone yang sama dengan instance Anda). Jika Anda memilih untuk menambahkan antarmuka jaringan tambahan yang berada di subnet VPC lain, Anda akan melihat opsi subnet multi-VPC saat memilih subnet. Jika Anda memilih subnet di VPC lain, label multi-VPC muncul di sebelah antarmuka jaringan yang telah Anda tambahkan. Ini memungkinkan Anda membuat instans multi-homed di seluruh VPC dengan konfigurasi jaringan dan keamanan yang berbeda. Perhatikan bahwa jika Anda melampirkan ENI tambahan dari VPC lain, Anda harus memilih grup keamanan untuk ENI dari VPC tersebut.

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#). Jika Anda menentukan lebih dari satu antarmuka jaringan, maka instans Anda tidak akan dapat menerima alamat IPv4 publik. Selain itu, jika Anda menentukan antarmuka jaringan yang ada untuk eth0, Anda tidak akan dapat mengganti pengaturan IPv4 publik subnet menggunakan Tetapkan Otomatis IP Publik. Untuk informasi selengkapnya, lihat [Menetapkan alamat IPv4 publik selama peluncuran instans](#).

Mengonfigurasi penyimpanan

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).

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 instans Amazon 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. Jika Anda telah memilih AMI dan instans yang memenuhi syarat untuk tingkat gratis, ingatlah bahwa agar tetap dalam tingkat gratis, Anda harus tetap di bawah 30 GiB dari total 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 tipe volume SSD IOPS yang Tersedia, maka Anda dapat memasukkan jumlah operasi I/O per detik (IOPS) yang dapat didukung oleh volume tersebut.
- 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.
- Sistem file: Pasang sistem file Amazon EFS atau Amazon FSx ke instans. Untuk informasi selengkapnya tentang pemasangan sistem file Amazon EFS, lihat [Menggunakan Amazon EFS dengan instans Linux](#). Untuk informasi selengkapnya tentang pemasangan sistem file Amazon FSx, lihat [Menggunakan Amazon FSx dengan Amazon EC2](#).

Detail lanjutan

Untuk Detail lanjutan, perluas bagian untuk melihat kolom dan menentukan parameter tambahan apa pun untuk instans.

- Opsi pembelian: Pilih Minta Instans Spot untuk meminta Instans Spot dengan harga Spot, yang tidak akan melebihi harga Sesuai Permintaan, dan pilih Sesuaikan 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 EC2 meluncurkan Instans Sesuai Permintaan secara default. Untuk informasi selengkapnya, lihat [Membuat permintaan Instans Spot](#).
- Direktori gabungan domain: Pilih AWS 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 domain yang bergabung dengan instans Linux, lihat Menggabungkan instans [EC2 Linux dengan mulus ke direktori AD AWS Microsoft Terkelola](#). Untuk informasi selengkapnya tentang domain yang bergabung dengan instans Windows, lihat Menggabungkan [instans Windows EC2 dengan mulus ke direktori AWS Microsoft AD yang Dikelola](#).
- Profil instans IAM: Pilih profil instance AWS Identity and Access Management (IAM) untuk dikaitkan dengan instance. 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 [Tipe nama host instans Amazon EC2](#).
- Nama Host DNS: Menentukan apakah permintaan DNS ke nama sumber daya atau nama IP (tergantung pada pilihan Anda untuk Tipe hostname) akan merespons dengan alamat IPv4 (catatan A), alamat IPv6 (catatan AAAA), atau keduanya. Untuk informasi selengkapnya, lihat [Tipe nama host instans Amazon EC2](#).
- Perilaku pematian: Pilih apakah instans harus berhenti atau diakhiri saat dimatikan. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh 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 [Aktifkan perlindungan pengakhiran](#).
- 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 [Pantau instans Anda menggunakan CloudWatch](#).

- GPU Elastis: Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.
- Inferensi Elastis: Akselerator inferensi elastis untuk dipasang ke instans CPU EC2 Anda. Untuk informasi selengkapnya, lihat [Bekerja dengan Amazon Elastic Inference](#) dalam Panduan Developer Amazon Elastic Inference.

Note

Mulai 15 April 2023, tidak AWS akan memasukkan pelanggan baru ke Amazon Elastic Inference (EI), dan akan membantu pelanggan saat ini memigrasikan beban kerja mereka ke opsi yang menawarkan harga dan kinerja yang lebih baik. Setelah 15 April 2023, pelanggan baru tidak akan dapat meluncurkan instans dengan akselerator Amazon EI di Amazon, Amazon ECS, atau SageMaker Amazon EC2. Namun, pelanggan yang telah menggunakan Amazon EI setidaknya sekali selama periode 30 hari terakhir dianggap sebagai pelanggan saat ini dan akan dapat terus menggunakan layanan ini.

- 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 [Instans performa yang dapat melonjak](#).
- Nama grup penempatan: Tentukan grup penempatan untuk meluncurkan instans. 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](#).
- 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”](#).
- 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 lebih lanjut, lihat [Instans Khusus](#) dan [Host Khusus](#)

- ID disk RAM: (Hanya berlaku untuk AMI paravirtual (PV)) Pilih disk RAM untuk instans. Jika Anda telah memilih kernel, Anda mungkin perlu memilih RAM disk tertentu dengan driver untuk mendukungnya.
- Id Kernel: (Hanya berlaku untuk AMI paravirtual (PV)) Pilih kernel untuk instans.
- Nitro Enclave: Memungkinkan Anda untuk 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.
- Metadata dapat diakses: Anda dapat mengaktifkan atau menonaktifkan akses ke metadata instans. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans untuk instans baru](#).
- Titik akhir IPv6 metadata: Anda dapat mengaktifkan instance untuk menggunakan alamat IPv6 IMDS untuk mengambil metadata instance. [fd00:ec2::254] Opsi ini hanya tersedia jika Anda meluncurkan [instance yang dibangun di atas Sistem AWS Nitro](#) ke dalam [subnet yang didukung IPv6 \(dual stack atau IPv6\)](#) saja). Untuk informasi selengkapnya tentang pengambilan metadata instans, lihat [Mengambil metadata instans](#).
- Metadata versi: Jika Anda mengaktifkan akses ke metadata instans, maka Anda dapat memilih untuk meminta penggunaan instans Metadata Service Versi 2 saat meminta metadata instans. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans untuk instans baru](#).
- Metadata response hop limit: Jika Anda mengaktifkan metadata instans, maka Anda dapat menyetel 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 [Mengizinkan akses ke tanda dalam metadata instans](#).
- 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, lihat [Jalankan perintah pada instans Amazon EC2 Anda](#)

[saat peluncuran](#). Untuk informasi selengkapnya tentang data pengguna untuk instance Windows, lihat [Bagaimana 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 instans, untuk membantu memastikan bahwa Anda mempertahankan jumlah instans yang benar untuk menangani permintaan pada aplikasi, Anda dapat memilih pertimbangan EC2 Auto Scaling untuk membuat templat 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 instans yang berada dalam grup Auto Scaling sebagai tidak sehat, instans tersebut secara otomatis dijadwalkan untuk diganti ketika diakhiri dan instans yang lain diluncurkan, dan Anda akan 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 Auto Scaling Amazon EC2.

- 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.

Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat [Pemecahan masalah peluncuran instans](#).

(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.

Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama

Anda dapat meluncurkan instans menggunakan wizard peluncuran instans lama hanya jika Wilayah Anda mendukung pengalaman peluncuran lama. Wizard peluncuran instans menentukan semua parameter peluncuran yang diperlukan untuk meluncurkan sebuah instans. Jika wizard peluncuran instans memberikan nilai default, Anda dapat menerima default atau menentukan nilai Anda sendiri. Anda harus menentukan AMI dan pasangan kunci untuk meluncurkan sebuah instans.

Untuk petunjuk menggunakan wizard peluncuran instans baru, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

 **Important**

Saat Anda meluncurkan sebuah instans yang tidak termasuk dalam [AWS Tingkat Gratis](#), Anda akan dikenai biaya untuk waktu instans tersebut berjalan, meskipun instans tetap idle.

Langkah-langkah untuk meluncurkan sebuah instans:

- [Memulai peluncuran instans](#)
- [Langkah 1: Pilih Amazon Machine Image \(AMI\)](#)
- [Langkah 2: Pilih Tipe Instans](#)
- [Langkah 3: Konfigurasi Detail Instans](#)
- [Langkah 4: Tambahkan Penyimpanan](#)
- [Langkah 5: Tambahkan Tanda](#)
- [Langkah 6: Konfigurasi Grup Keamanan](#)
- [Langkah 7: Tinjau Peluncuran Instans dan Pilih Pasangan Kunci](#)

Memulai peluncuran instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di bilah navigasi di bagian atas layar, Wilayah saat ini ditampilkan (misalnya, AS Timur (Ohio)). Pilih Wilayah untuk instans yang memenuhi kebutuhan Anda. Pilihan ini penting karena beberapa sumber daya Amazon EC2 dapat dibagikan di antara Wilayah, sedangkan sumber daya yang lainnya tidak. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.

Langkah 1: Pilih Amazon Machine Image (AMI)

Saat Anda meluncurkan sebuah instans, Anda harus memilih konfigurasi, yang dikenal sebagai Amazon Machine Image (AMI). AMI berisi informasi yang diperlukan untuk membuat instans baru. Misalnya, AMI mungkin berisi perangkat lunak yang diperlukan untuk bertindak sebagai server web, seperti Linux, Apache, dan situs web Anda.

Saat Anda meluncurkan sebuah instans, Anda dapat memilih AMI dari daftar, atau Anda dapat memilih parameter Systems Manager yang mengarah ke ID AMI. Untuk informasi selengkapnya, lihat [the section called “Temukan AMI menggunakan parameter Systems Manager”](#).

Di halaman Pilih Amazon Machine Image (AMI), gunakan salah satu dari dua opsi untuk memilih AMI. [Cari di daftar AMI](#), atau [Cari berdasarkan parameter Systems Manager](#).

Dengan mencari daftar AMI

1. Pilih jenis AMI yang akan digunakan di panel kiri:

Mulai Cepat

Pemilihan AMI populer untuk membantu Anda memulai dengan cepat. Untuk memilih AMI yang memenuhi syarat untuk tingkat gratis, pilih Tingkat gratis saja di panel kiri. AMI tersebut ditandai sebagai Memenuhi Syarat Tingkat Gratis.

AMI saya

AMI privat yang Anda miliki, atau AMI privat yang telah dibagikan dengan Anda. Untuk melihat AMI yang dibagikan dengan Anda, pilih Dibagikan dengan saya di panel kiri.

AWS Marketplace

Toko online tempat Anda dapat membeli perangkat lunak yang berjalan AWS, termasuk AMI. Untuk informasi selengkapnya tentang meluncurkan instance dari AWS Marketplace, lihat [Luncurkan sebuah AWS Marketplace instance](#).

AMI Komunitas

AMI yang telah disediakan oleh anggota AWS komunitas untuk digunakan orang lain. Untuk mem-filter daftar AMI berdasarkan sistem operasi, pilih kotak centang yang sesuai pada Sistem Operasi. Anda juga dapat memfilter berdasarkan arsitektur dan tipe perangkat root.

2. (Instance Linux) Periksa jenis perangkat Root yang terdaftar untuk setiap AMI. Perhatikan tipe AMI yang Anda butuhkan, baik `ebs` (didukung oleh Amazon EBS) atau `instance-store` (didukung oleh penyimpanan instans). Untuk informasi selengkapnya, lihat [Penyimpanan untuk perangkat root](#).
3. Periksa Tipe virtualisasi yang tercantum untuk setiap AMI. Perhatikan AMI tipe yang mana yang Anda butuhkan, baik `hvm` atau `paravirtual`. Sebagai contoh, beberapa tipe instans memerlukan HVM. Untuk informasi selengkapnya tentang jenis virtualisasi Linux, lihat [Tipe virtualisasi AMI](#).
4. Periksa Mode booting yang terdaftar untuk setiap AMI. Perhatikan AMI yang mana yang menggunakan mode booting yang Anda butuhkan, baik `legacy-bios` atau `uefi`. Untuk informasi selengkapnya, lihat [Mode boot Amazon EC2](#).
5. Pilih AMI yang memenuhi kebutuhan Anda, lalu pilih Pilih.

Dengan parameter Systems Manager

1. Pilih Cari berdasarkan parameter Systems Manager (di kanan atas).
2. Untuk Parameter System Manager, pilih parameter. ID AMI terkait muncul di samping Saat ini menyelesaikan.
3. Pilih Cari. AMI yang cocok dengan ID AMI muncul dalam daftar.
4. Pilih AMI dari daftar, lalu pilih Pilih.

Langkah 2: Pilih Tipe Instans

Di halaman Pilih Tipe Instans, pilih konfigurasi perangkat keras dan ukuran instans yang akan diluncurkan. Tipe instans yang lebih besar memiliki lebih banyak CPU dan memori. Untuk informasi selengkapnya, lihat [Jenis Instans Amazon EC2](#).

Agar tetap memenuhi syarat untuk tingkat gratis, pilih tipe instans t2.micro (atau tipe instans t3.micro di Wilayah di mana t2.micro tidak tersedia). Jika tipe instans memenuhi syarat untuk masuk Tingkat Gratis, instans tersebut diberi label Memenuhi syarat Tingkat Gratis. Untuk informasi selengkapnya tentang t2.micro dan t3.micro, lihat [Instans performa yang dapat melonjak](#).

Secara default, wizard menampilkan tipe instans generasi saat ini, dan memilih tipe instans pertama yang tersedia berdasarkan AMI yang Anda pilih. Untuk melihat tipe instans generasi sebelumnya, pilih Semua generasi dari daftar filter.

Note

Untuk menyiapkan instans dengan cepat untuk tujuan pengujian, pilih Tinjau dan Luncurkan untuk menerima pengaturan konfigurasi default, dan luncurkan instans Anda. Atau, untuk mengonfigurasi instans Anda lebih lanjut, pilih Berikutnya: Konfigurasi Detail Instans.

Langkah 3: Konfigurasi Detail Instans

Pada halaman Konfigurasi Detail Instans, ubah pengaturan berikut seperlunya (perluas Detail Lanjutan untuk melihat semua pengaturan), lalu pilih Berikutnya: Tambah Penyimpanan:

- Jumlah instans: Masukkan jumlah instans yang akan diluncurkan.

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) Untuk membantu memastikan bahwa Anda mempertahankan jumlah instans yang benar untuk menangani permintaan pada aplikasi, Anda dapat memilih Luncurkan ke Grup Auto Scaling untuk membuat konfigurasi 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 instans yang berada dalam grup Auto Scaling sebagai tidak sehat, instans tersebut secara otomatis dijadwalkan untuk diganti ketika diakhiri dan instans yang lain diluncurkan, dan Anda akan 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 [Pemeriksaan kondisi untuk instans Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

- Opsi pembelian: Pilih Minta instans Spot untuk meluncurkan Instans Spot. Ini menambah dan menghapus opsi dari halaman ini. Anda dapat secara opsional mengatur harga maksimum Anda (tidak disarankan), dan secara opsional mengubah tipe permintaan, perilaku interupsi, dan validitas permintaan. Untuk informasi selengkapnya, lihat [Membuat permintaan Instans Spot](#).
- Jaringan: Pilih VPC, atau untuk membuat VPC baru, pilih Buat VPC baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke wizard peluncuran instans dan pilih Segarkan untuk memuat VPC Anda dalam daftar.
- Subnet: Anda dapat meluncurkan sebuah instans di subnet yang terkait dengan Zona Ketersediaan, Zona Lokal, Zona Wavelength, atau Outpost.

Untuk meluncurkan instans di Zonz Ketersediaan, pilih subnet tempat Anda meluncurkan instans. Anda dapat memilih Tidak ada preferensi untuk membiarkan AWS memilih subnet default di Availability Zone apa pun. Untuk membuat subnet baru, pilih Buat subnet baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke wizard dan pilih ikon Segarkan untuk memuat subnet Anda dalam daftar.

Untuk meluncurkan instans di Local Zone, pilih subnet yang Anda buat di Local Zone.

Untuk meluncurkan sebuah instans di Outpost, pilih subnet di VPC yang Anda kaitkan dengan sebuah Outpost.

- Tetapkan otomatis IP Publik: Tentukan apakah instans Anda menerima alamat IPv4 publik yang lain. Secara default, instans di subnet default menerima alamat IPv4 publik, sedangkan instans di subnet nondefault tidak menerimanya. Anda dapat memilih Aktifkan atau Nonaktifkan untuk mengganti pengaturan default subnet. Untuk informasi selengkapnya, lihat [Alamat IPv4 publik](#).
- Tetapkan otomatis IP IPv6: Tentukan apakah instans Anda menerima alamat IPv6 dari rentang subnet. Pilih Aktifkan atau Nonaktifkan untuk mengganti pengaturan default subnet. Opsi ini

hanya tersedia jika Anda telah mengaitkan Blok CIDR IPv6 dengan VPC dan subnet Anda. Untuk informasi selengkapnya, lihat [Menambahkan blok CIDR IPv6 ke VPC](#) di Panduan Pengguna Amazon VPC.

- Jenis nama host: Pilih apakah nama host OS tamu dari instans akan menyertakan nama sumber daya atau nama IP. Untuk informasi selengkapnya, lihat [Tipe nama host instans Amazon EC2](#).
- Nama Host DNS: Menentukan apakah permintaan DNS ke nama sumber daya atau nama IP (tergantung pada pilihan Anda untuk Tipe hostname) akan merespons dengan alamat IPv4 (catatan A), alamat IPv6 (catatan AAAA), atau keduanya. Untuk informasi selengkapnya, lihat [Tipe nama host instans Amazon EC2](#).
- Direktori gabungan domain: Pilih AWS 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 domain yang bergabung dengan instans Linux, lihat [Menggabungkan instans EC2 Linux dengan mulus ke direktori AD AWS Microsoft Terkelola](#). Untuk informasi selengkapnya tentang domain yang bergabung dengan instans Windows [Bergabunglah dengan instans EC2 Windows dengan mulus](#).
- Grup penempatan: Grup penempatan menentukan strategi penempatan instans Anda. Pilih grup penempatan yang ada, atau buat yang baru. Opsi ini hanya tersedia jika Anda telah memilih tipe instans yang mendukung grup penempatan. Untuk informasi selengkapnya, lihat [Grup penempatan](#).
- Reservasi Kapasitas: Tentukan apakah akan meluncurkan instans ke dalam kapasitas bersama, Reservasi Kapasitas open, Reservasi Kapasitas tertentu, atau grup Reservasi Kapasitas. Untuk informasi selengkapnya, lihat [Luncurkan instans ke dalam Reservasi Kapasitas yang ada](#).
- Peran IAM: Pilih peran AWS Identity and Access Management (IAM) untuk dikaitkan dengan instance. Untuk informasi selengkapnya, lihat [IAM role untuk Amazon EC2](#).
- Opsi CPU: Pilih Tentukan opsi CPU untuk menentukan jumlah vCPU kustom selama peluncuran. Atur jumlah inti CPU dan thread per inti. Untuk informasi selengkapnya, lihat [Mengoptimalkan opsi CPU](#).
- Perilaku pematian: Pilih apakah instans harus berhenti atau diakhiri saat dimatikan. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh instans](#).
- Berhenti - Perilaku hibernasi: Untuk mengaktifkan hibernasi, pilih kotak centang ini. Opsi ini hanya tersedia jika instans Anda memenuhi prasyarat hibernasi. Untuk informasi selengkapnya, lihat [Hibernasi instans Amazon EC2 Anda](#).
- Aktifkan proteksi terminasi: Untuk mencegah penghentian yang tidak disengaja, pilih kotak centang ini. Untuk informasi selengkapnya, lihat [Aktifkan perlindungan pengakhiran](#).

- **Aktifkan perlindungan penghentian:** Untuk mencegah penghentian yang tidak disengaja, pilih kotak centang ini. Untuk informasi selengkapnya, lihat [Aktifkan perlindungan penghentian](#).
- **Pemantauan:** Pilih kotak centang ini untuk mengaktifkan pemantauan mendetail instans Anda menggunakan Amazon CloudWatch. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat [Pantau instans Anda menggunakan CloudWatch](#).
- **Instans dengan pengoptimalan EBS:** Instans dengan pengoptimalan Amazon EBS menggunakan tumpukan konfigurasi yang dioptimalkan dan menyediakan kapasitas khusus tambahan untuk I/O Amazon EBS. Jika tipe instans mendukung fitur ini, pilih kotak centang ini untuk mengaktifkannya. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat [Instans yang dioptimalkan Amazon EBS](#).
- **Penghunian:** Jika Anda meluncurkan instans Anda ke VPC, Anda dapat memilih untuk menjalankan instans Anda pada perangkat keras terisolasi dan khusus (Khusus) atau di Host Khusus (Host khusus). Biaya tambahan mungkin berlaku. Untuk informasi lebih lanjut, lihat [Instans Khusus](#) dan [Host Khusus](#).
- **T2/T3 Tak Terbatas:** Pilih kotak centang ini untuk mengaktifkan aplikasi agar melonjak melampaui acuan selama diperlukan. Biaya tambahan mungkin berlaku. Untuk informasi selengkapnya, lihat [Instans performa yang dapat melonjak](#).
- **Sistem File:** Untuk membuat sistem file baru untuk dipasang ke instans Anda, pilih **Buat sistem file baru**, masukkan nama untuk sistem file baru tersebut, lalu pilih **Buat**. Sistem file dibuat menggunakan Amazon EFS Quick Create, yang menerapkan pengaturan yang direkomendasikan layanan. Grup keamanan yang diperlukan untuk mengaktifkan akses ke sistem file secara otomatis dibuat dan dilampirkan ke instans dan target pemasangan dari sistem file. Anda juga dapat memilih untuk membuat dan melampirkan grup keamanan yang diperlukan secara manual. Untuk memasang satu atau beberapa sistem file Amazon EFS yang ada ke instans Anda, pilih **Tambahkan sistem file** lalu pilih sistem file yang akan dipasang dan titik pemasangan yang akan digunakan. Untuk informasi selengkapnya, lihat [Menggunakan Amazon EFS dengan instans Linux](#).
- **Antarmuka Jaringan:** Jika Anda memilih subnet tertentu, maka Anda dapat menentukan hingga dua antarmuka jaringan untuk instans Anda:
 - Untuk **Network Interface**, pilih **Antarmuka jaringan baru** untuk memungkinkan AWS membuat antarmuka baru, atau pilih antarmuka jaringan yang ada dan tersedia.
 - Untuk **IP Primer**, masukkan alamat IPv4 pribadi dari kisaran subnet Anda, atau tinggalkan **Auto-assign** untuk mengizinkan AWS memilih alamat IPv4 pribadi untuk Anda.
 - Untuk **alamat IP Sekunder**, pilih **Tambahkan IP** untuk menetapkan lebih dari satu alamat IPv4 privat ke antarmuka jaringan yang dipilih.

- (Hanya IPv6) Untuk IPv6 IP, pilih Tambah IP, dan masukkan alamat IPv6 dari rentang subnet, atau biarkan Auto-assign untuk membiarkan memilih satu untuk Anda. AWS
- Indeks Kartu Jaringan: Indeks kartu jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0. Beberapa tipe instans mendukung banyak kartu jaringan.
- Pilih Tambahkan Perangkat untuk menambahkan antarmuka jaringan sekunder. Antarmuka jaringan sekunder dapat berada di subnet VPC yang berbeda, selama masih berada di Zona Ketersediaan yang sama dengan instans Anda.

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#). Jika Anda menentukan lebih dari satu antarmuka jaringan, maka instans Anda tidak akan dapat menerima alamat IPv4 publik. Selain itu, jika Anda menentukan antarmuka jaringan yang ada untuk eth0, Anda tidak akan dapat mengganti pengaturan IPv4 publik subnet menggunakan Tetapkan Otomatis IP Publik. Untuk informasi selengkapnya, lihat [Menetapkan alamat IPv4 publik selama peluncuran instans](#).

- ID Kernel: (Hanya berlaku untuk AMI paravirtual (PV)) Pilih Gunakan default kecuali Anda ingin menggunakan kernel tertentu.
- ID RAM disk: (Hanya berlaku untuk AMI paravirtual (PV)) Pilih Gunakan default kecuali Anda ingin menggunakan RAM disk tertentu. Jika Anda telah memilih kernel, Anda mungkin perlu memilih RAM disk tertentu dengan driver untuk mendukungnya.
- Enclave: 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.
- Metadata dapat diakses: Anda dapat mengaktifkan atau menonaktifkan akses ke Layanan Metadata Instans (IMDS). Untuk informasi selengkapnya, lihat [Gunakan IMDSv2](#).
- Titik akhir IPv6 metadata: Anda dapat mengaktifkan instance untuk menggunakan alamat IPv6 IMDS untuk mengambil metadata instance. [fd00:ec2::254] Opsi ini hanya tersedia jika Anda meluncurkan [instance yang dibangun di atas Sistem AWS Nitro](#) ke dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#). Untuk informasi selengkapnya tentang pengambilan metadata instans, lihat [Mengambil metadata instans](#).
- 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 token metadata: Jika Anda mengaktifkan IMDS, maka Anda dapat mengatur jumlah lompatan jaringan yang diizinkan untuk token metadata. Untuk informasi selengkapnya, lihat [Gunakan IMDSv2](#).

- **Data pengguna:** Anda dapat menentukan data pengguna untuk mengonfigurasi instans selama peluncuran, atau untuk menjalankan skrip konfigurasi. Untuk melampirkan file, pilih opsi Sebagai file dan telusuri file yang akan dilampirkan.

Langkah 4: Tambahkan Penyimpanan

AMI yang Anda pilih mencakup satu atau lebih volume penyimpanan, termasuk volume perangkat root. Di halaman Add Storage, Anda dapat menentukan volume tambahan untuk dilampirkan ke instans dengan memilih Add New Volume. Konfigurasi setiap volume sebagai berikut, lalu pilih Berikutnya: Tambahkan Tanda.

- **Tipe:** Pilih volume penyimpanan instans atau Amazon EBS untuk dikaitkan dengan instans Anda. Tipe volume yang tersedia di daftar tergantung pada tipe instans yang telah Anda pilih. Untuk informasi selengkapnya, lihat [Penyimpanan instans Amazon EC2](#) dan [volume Amazon EBS](#).
- **Perangkat:** Pilih dari daftar nama perangkat yang tersedia untuk volume.
- **Snapshot:** Masukkan nama atau ID dari snapshot yang akan mengembalikan volume. Anda juga dapat mencari snapshot bersama dan publik yang tersedia dengan mengetik teks ke dalam bidang Snapshot. Deskripsi snapshot peka huruf besar kecil.
- **Ukuran:** Untuk volume EBS, Anda dapat menentukan ukuran penyimpanan. Meskipun Anda telah memilih AMI dan instans yang memenuhi syarat untuk tingkat gratis, untuk tetap dalam tingkat gratis, Anda harus tetap di bawah 30 GiB dari total 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 tipe volume SSD IOPS yang Tersedia, maka Anda dapat memasukkan jumlah operasi I/O per detik (IOPS) yang dapat didukung oleh volume tersebut.
- **Hapus saat Pengakhiran:** Untuk volume Amazon EBS, pilih kotak centang ini untuk menghapus volume saat instans diakhiri. Untuk informasi selengkapnya, lihat [Pertahankan data saat instans diakhiri](#).
- **Terenkripsi:** Jika tipe instans mendukung enkripsi EBS, maka Anda dapat menentukan status enkripsi 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 lain atau menonaktifkan enkripsi. Untuk informasi selengkapnya, lihat [enkripsi Amazon EBS](#) di Panduan Pengguna Amazon EBS.

Langkah 5: Tambahkan Tanda

Pada halaman Tambahkan Tanda, tentukan [tanda](#) dengan memberikan kombinasi kunci dan nilai. Anda dapat menandai instans, volume, atau keduanya. Untuk Instans Spot, Anda hanya dapat menandai permintaan Instans Spot. Pilih Tambahkan tanda lain untuk menambahkan lebih dari satu tanda ke sumber daya Anda. Pilih Berikutnya: Konfigurasi Grup Keamanan setelah Anda selesai.

Langkah 6: Konfigurasi Grup Keamanan

Di halaman Konfigurasi Grup Keamanan, gunakan grup keamanan untuk menentukan aturan firewall bagi instans Anda. Aturan ini menentukan lalu lintas jaringan yang masuk yang dikirim ke instans Anda. Semua lalu lintas lainnya diabaikan. (Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).) Pilih atau buat grup keamanan sebagai berikut, lalu pilih Tinjau dan Luncurkan.

- Untuk memilih grup keamanan yang ada, pilih Pilih grup keamanan yang ada, dan pilih grup keamanan. Anda tidak bisa mengedit aturan grup keamanan yang sudah ada, tapi Anda bisa menyalinnya ke grup baru dengan memilih Salin ke baru. Kemudian Anda dapat menambahkan aturan seperti yang dijelaskan di langkah berikutnya.
- Untuk membuat grup keamanan baru, pilih Buat grup keamanan baru. Wizard secara otomatis mendefinisikan grup keamanan launch-wizard- x dan membuat aturan masuk untuk memungkinkan Anda terhubung ke instance Anda. Instance Linux menggunakan aturan masuk untuk SSH (port 22) dan instance Windows menggunakan aturan masuk untuk RDP (port 3389).
- Anda dapat menambahkan aturan untuk menyesuaikan dengan kebutuhan Anda. Misalnya, jika instans Anda adalah server web, buka port 80 (HTTP) dan 443 (HTTPS) untuk mengizinkan lalu lintas internet.

Untuk menambahkan aturan, pilih Tambahkan Aturan, pilih protokol yang akan dibuka untuk lalu lintas jaringan, lalu tentukan sumbernya. Pilih IP Saya dari daftar Sumber agar wizard menambahkan alamat IP publik komputer Anda. Jika Anda terhubung melalui ISP atau dari belakang firewall Anda tanpa alamat IP statis, maka Anda harus menemukan rentang alamat IP yang digunakan oleh komputer klien.

Warning

Aturan yang memungkinkan semua alamat IP (0.0.0.0/0) untuk mengakses instans Anda melalui SSH atau RDP dapat diterima untuk latihan singkat ini, tetapi tidak aman

untuk lingkungan produksi. Anda hanya boleh mengotorisasi alamat IP atau rentang alamat tertentu saja untuk mengakses instans.

Langkah 7: Tinjau Peluncuran Instans dan Pilih Pasangan Kunci

Pada halaman Tinjau Peluncuran Instans, periksa detail dari instans, dan buat perubahan yang diperlukan dengan memilih tautan Edit yang sesuai.

Saat Anda siap, pilih Luncurkan.

Dalam kotak dialog Pilih pasangan kunci yang sudah ada atau buat pasangan kunci baru, Anda dapat memilih pasangan kunci yang sudah ada, atau membuat yang baru. Misalnya, pilih Pilih pasangan kunci yang ada, lalu pilih pasangan kunci yang Anda buat saat menyiapkan. Untuk informasi selengkapnya, lihat [Pasangan kunci Amazon EC2 dan instans Amazon EC2](#).

Important

Jika Anda memilih opsi Lanjutkan tanpa pasangan kunci, Anda tidak akan dapat terhubung ke instans, kecuali Anda memilih AMI yang dikonfigurasi agar pengguna dapat masuk dengan cara lain.

Untuk meluncurkan instans Anda, centang kotak penerimaan, lalu pilih Luncurkan Instans.

(Opsional) Anda dapat membuat alarm pemeriksaan status untuk instans tersebut (dapat dikenai biaya tambahan). Di layar konfirmasi, pilih Membuat alarm pemeriksaan status dan ikuti petunjuknya. Alarm pemeriksaan status juga dapat dibuat setelah Anda meluncurkan instans. Untuk informasi selengkapnya, lihat [Membuat dan mengedit alarm pemeriksaan status](#).

Jika instance gagal diluncurkan atau status langsung beralih ke `terminated` dari pada `running`, lihat [Pemecahan masalah peluncuran instans](#).

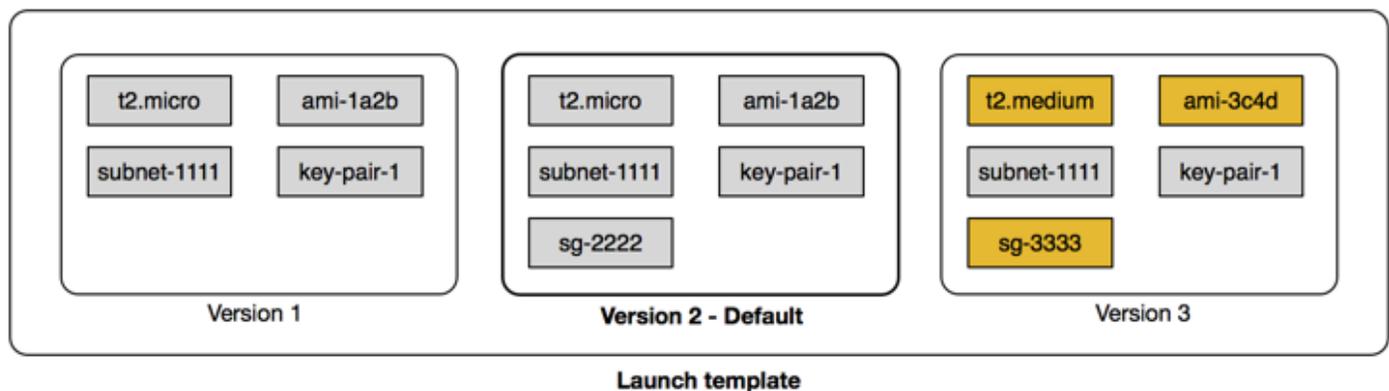
Meluncurkan sebuah instans dari templat peluncuran

Anda dapat menggunakan template peluncuran untuk menyimpan parameter peluncuran instance sehingga Anda tidak perlu menentukannya setiap kali Anda meluncurkan instance. Misalnya, Anda dapat membuat template peluncuran dengan ID AMI, jenis instans, dan setelan jaringan yang biasanya Anda gunakan untuk meluncurkan instance. Saat meluncurkan instance menggunakan

konsol Amazon EC2, AWS SDK, atau alat baris perintah, Anda dapat menentukan templat 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 akan 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

- [Larangan templat peluncuran](#)
- [Kontrol akses untuk templat peluncuran dengan izin IAM](#)
- [Gunakan templat peluncuran untuk mengontrol instans peluncuran](#)
- [Membuat templat peluncuran](#)
- [Modifikasi templat peluncuran \(mengelola versi templat peluncuran\)](#)
- [Hapus templat peluncuran](#)
- [Meluncurkan instans dari templat peluncuran](#)

Larangan templat peluncuran

Aturan berikut berlaku untuk templat peluncuran dan versi templat peluncuran:

- **Kuota** - Untuk melihat kuota untuk template peluncuran dan meluncurkan versi template, buka konsol [Service Quotas](#) atau gunakan perintah. [list-service-quotas](#) AWS CLI Setiap AWS akun dapat memiliki hingga maksimum 5.000 templat peluncuran per Wilayah dan hingga 10.000 versi per templat 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 Anda untuk meluncurkan sebuah instans mencakup semua parameter yang diperlukan. Misalnya, jika templat peluncuran Anda tidak menyertakan ID AMI, Anda harus menentukan templat peluncuran dan ID AMI saat Anda meluncurkan sebuah instans.
- **Parameter tidak divalidasi** — Parameter templat peluncuran tidak sepenuhnya divalidasi saat Anda membuat templat peluncuran. Jika Anda menentukan nilai yang salah untuk parameter, atau jika Anda tidak menggunakan kombinasi parameter yang didukung, tidak ada instans yang dapat diluncurkan menggunakan templat peluncuran ini. Pastikan Anda menentukan nilai yang benar untuk parameter dan Anda menggunakan kombinasi parameter yang didukung. Misalnya, untuk meluncurkan sebuah instans di grup penempatan, Anda harus menentukan tipe instans yang didukung.
- **Tanda** — Anda dapat memberi tanda pada templat peluncuran, tetapi Anda tidak dapat memberi tanda pada versi templat 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 Anda membuat versi templat peluncuran, Anda tidak dapat menentukan nomor versi sendiri.

Kontrol akses untuk templat peluncuran dengan izin IAM

Anda dapat menggunakan izin IAM untuk mengontrol tindakan templat peluncuran yang dapat dilakukan pengguna, seperti melihat, membuat, atau menghapus templat peluncuran.

Jika Anda memberikan izin kepada pengguna untuk membuat templat peluncuran dan meluncurkan versi templat, Anda tidak dapat menggunakan izin tingkat sumber daya untuk membatasi sumber daya yang dapat mereka tentukan dalam templat peluncuran. Oleh karena itu, pastikan Anda memberikan izin untuk membuat templat peluncuran dan meluncurkan versi templat hanya kepada administrator yang sesuai.

Anda harus memberi siapa pun yang akan menggunakan template peluncuran izin yang diperlukan untuk membuat dan mengakses sumber daya yang ditentukan dalam template peluncuran. Sebagai contoh:

- Untuk meluncurkan instance dari Amazon Machine Image (AMI) pribadi bersama, pengguna harus memiliki izin peluncuran untuk AMI.
- Untuk membuat volume EBS dengan tag dari snapshot yang ada, pengguna harus memiliki akses baca ke snapshot, dan izin untuk membuat dan menandai volume.

Daftar Isi

- [ec2: Templat CreateLaunch](#)
- [ec2: Template DescribeLaunch](#)
- [EC2: DescribeLaunch TemplateVersions](#)
- [ec2: Templat DeleteLaunch](#)
- [Mengontrol izin versioning](#)
- [Kontrol akses ke tanda pada templat peluncuran](#)

ec2: Templat CreateLaunch

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, `CreateLaunchTemplate` penelepon harus memiliki `ec2:CreateTags` izin dalam kebijakan IAM.
- `ec2:RunInstances` — Untuk meluncurkan instans EC2 dari template peluncuran yang mereka buat, kepala sekolah 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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#).

```

{
  "Sid": "IAMPolicyForTaggingLaunchTemplatesOnCreation",
  "Action": "ec2:CreateTags",
  "Effect": "Allow",
  "Resource": "arn:aws:ec2:region:account-id: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: Template DescribeLaunch

Untuk mendaftar templat peluncuran di akun, pengguna utama harus memiliki izin `ec2:DescribeLaunchTemplates` 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 kepada pengguna utama untuk menampilkan daftar semua templat peluncuran di akun.

```
{
  "Sid": "IAMPolicyForDescribingLaunchTemplates",
  "Action": "ec2:DescribeLaunchTemplates",
  "Effect": "Allow",
  "Resource": "*"
}
```

EC2: DescribeLaunch TemplateVersions

Pengguna utama yang melihat templat peluncuran juga harus memiliki izin `ec2:DescribeLaunchTemplateVersions` untuk mengambil seluruh rangkaian atribut yang membentuk templat peluncuran.

Untuk mendaftar versi templat peluncuran di akun, pengguna utama harus memiliki izin `ec2:DescribeLaunchTemplateVersions` 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 kepada pengguna utama untuk menampilkan daftar semua versi templat peluncuran di akun.

```
{
  "Sid": "IAMPolicyForDescribingLaunchTemplateVersions",
  "Effect": "Allow",
  "Action": "ec2:DescribeLaunchTemplateVersions",
  "Resource": "*"
}
```

ec2: Templat DeleteLaunch

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 kepada pengguna utama untuk menghapus templat peluncuran hanya jika templat menggunakan tanda yang ditentukan (*`purpose=testing`*).

```
{
  "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:region:account-id: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 templat peluncuran.

- Saat membuat versi templat peluncuran, Anda memengaruhi AWS sumber daya apa pun yang memungkinkan Amazon EC2 meluncurkan instans atas nama Anda dengan versi tersebut. Latest
- Saat Anda memodifikasi templat peluncuran, Anda dapat mengubah versi mana yang merupakan versi 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 perlu berhati-hati dalam menangani AWS sumber daya yang berinteraksi dengan Latest atau Default meluncurkan versi template, seperti Armada EC2 dan Armada Spot. 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 dengan 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: DeleteLaunch TemplateVersion

Important

Seperti biasa, Anda harus berhati-hati saat memberikan izin kepada pengguna utama untuk menghapus sumber daya. Menghapus versi template peluncuran dapat menyebabkan kegagalan pada 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:region:account-id: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 untuk mengontrol instans peluncuran

Anda dapat 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

Templat peluncuran dapat berisi semua atau beberapa parameter untuk meluncurkan sebuah instans. Saat Anda meluncurkan sebuah instans menggunakan templat peluncuran, Anda dapat mengganti parameter yang ditentukan di templat peluncuran. Atau, Anda dapat menentukan parameter tambahan yang tidak ada di templat peluncuran.

Note

Anda tidak dapat menghapus parameter templat 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 instans, pengguna harus memiliki izin untuk menggunakan tindakan `ec2:RunInstances`. 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 di templat 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

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](#).

Membuat templat peluncuran

Buat template peluncuran menggunakan parameter yang Anda tentukan, atau gunakan templat peluncuran yang ada atau instance sebagai dasar untuk templat peluncuran baru.

Tugas

- [Buat template peluncuran dari 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 dari parameter

Untuk membuat templat peluncuran, Anda harus menentukan nama templat peluncuran dan setidaknya satu parameter konfigurasi instans.

Arah konsol

Untuk membuat template peluncuran menggunakan konsol

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. Parameter template peluncuran dikelompokkan. Untuk detail tentang setiap grup, lihat bagian di bawah ini.
4. Gunakan panel Summary untuk meninjau konfigurasi template peluncuran Anda. Anda dapat menavigasi ke bagian mana pun dengan memilih tautannya dan kemudian membuat perubahan yang diperlukan.
5. Ketika Anda siap untuk membuat templat peluncuran Anda, pilih Buat templat peluncuran.

Luncurkan nama template, deskripsi, dan tanda

1. Untuk Nama templat peluncuran, masukkan nama deskriptif untuk templat peluncuran.
2. Untuk Deskripsi versi templat, berikan deskripsi singkat tentang versi templat peluncuran ini.
3. Untuk memberi [tanda](#) pada templat peluncuran saat pembuatan, perluas Tanda templat, pilih Tambahkan tag, lalu masukkan kunci tanda dan pasangan nilai. Pilih Tambah tanda 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 [Tanda sumber daya](#).

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 tempat peluncuran tanpa memilih AMI.

Bilah pencarian

Untuk mencari melalui semua AMI yang tersedia, masukkan kata kunci di bilah pencarian AMI dan kemudian tekan Enter. Untuk memilih AMI, pilih Pilih.

Terbaru

AMI yang baru saja Anda gunakan.

Pilih Baru diluncurkan atau Saat ini sedang digunakan, kemudian pilih AMI dari Amazon Machine Image (AMI).

AMI saya

AMI privat yang Anda miliki, atau AMI privat yang telah dibagikan dengan Anda.

Pilih Milik saya atau Dibagikan dengan saya, kemudian pilih AMI dari Amazon Machine Image (AMI).

Mulai Cepat

AMI 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 memenuhi syarat untuk tingkat gratis, pastikan bahwa AMI ditandai dengan Tingkat gratis yang memenuhi syarat.

Telusuri AMI lainnya

Pilih Telusuri AMI lainnya untuk menelusuri katalog lengkap AMI.

- Untuk menelusuri semua AMI yang tersedia, masukkan kata kunci di bilah pencarian kemudian tekan Enter.
- Untuk menemukan AMI menggunakan parameter Systems Manager, pilih tombol panah di sebelah kanan bilah pencarian, lalu pilih parameter Cari berdasarkan Systems Manager. Untuk informasi selengkapnya, lihat [Temukan AMI menggunakan parameter Systems Manager](#).
- Untuk menentukan parameter Systems Manager yang akan menyelesaikan AMI pada saat instans diluncurkan dari templat peluncuran, pilih tombol panah di sebelah kanan bilah pencarian, lalu pilih Tentukan parameter nilai kustom/Systems Manager. Untuk informasi selengkapnya, lihat [Gunakan parameter Systems Manager alih-alih ID AMI](#).
- Untuk mencari berdasarkan kategori, pilih AMI Mulai Cepat, AMI Saya, AMI AWS Marketplace , atau AMI Komunitas.

AWS Marketplace Ini adalah toko online tempat Anda dapat membeli perangkat lunak yang berjalan AWS, termasuk AMI. Untuk informasi selengkapnya tentang meluncurkan instance dari AWS Marketplace, lihat [Luncurkan sebuah AWS Marketplace instance](#). Di AMI Komunitas, Anda dapat menemukan AMI yang telah disediakan oleh anggota AWS komunitas untuk digunakan orang lain. AMI dari Amazon atau mitra terverifikasi ditandai sebagai Penyedia terverifikasi.

- Untuk memfilter daftar AMI, 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 mana tipe AMI yang Anda butuhkan: baik ebs (didukung oleh Amazon EBS) atau instance-store (didukung oleh penyimpanan instans. Untuk informasi selengkapnya, lihat [Penyimpanan untuk perangkat root](#).
- Periksa tipe Virtualisasi yang terdaftar untuk setiap AMI. Perhatikan mana tipe AMI yang Anda butuhkan, baik hvm atau paravirtual. Sebagai contoh, beberapa tipe instans memerlukan HVM. Untuk informasi selengkapnya, lihat [Tipe virtualisasi AMI](#).
- Periksa Mode booting yang terdaftar untuk setiap AMI. Perhatikan mana AMI yang menggunakan mode booting yang Anda butuhkan: baik legacy-bios, uefi, atau uefi-preferred. Untuk informasi selengkapnya, lihat [Mode boot Amazon EC2](#).
- Pilih AMI yang memenuhi kebutuhan Anda, lalu pilih Pilih.

Jenis 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 [instans Amazon EC2](#).

Untuk Tipe instans, Anda dapat memilih tipe instans, atau Anda dapat menentukan atribut instans dan membiarkan Amazon EC2 mengidentifikasi tipe instans dengan atribut tersebut.

Note

Menentukan atribut instans hanya didukung saat menggunakan grup Auto Scaling, Armada EC2, dan Armada Spot untuk meluncurkan instans. Untuk informasi selengkapnya, lihat [Membuat grup Auto Scaling menggunakan pemilihan tipe instans berbasis atribut](#), [Pemilihan tipe instans berbasis atribut untuk Armada EC2](#), dan [Pemilihan tipe instans berbasis atribut untuk Armada Spot](#).

Jika Anda berencana untuk menggunakan template peluncuran di [wizard instance peluncuran](#) atau dengan [RunInstancesAPI](#), Anda harus memilih jenis instance.

- Tipe Instans: Pastikan bahwa tipe instans kompatibel dengan AMI yang Anda tentukan. Untuk informasi selengkapnya, lihat [Jenis Instans Amazon EC2](#).
- Bandingkan tipe instans: Anda dapat membandingkan tipe instans yang berbeda dengan atribut berikut: jumlah vCPU, arsitektur, jumlah memori (GiB), jumlah penyimpanan (GB), tipe penyimpanan, dan performa jaringan.
- Dapatkan saran: Anda bisa mendapatkan panduan dan saran mengenai tipe instans dari pemilih tipe instans Amazon Q EC2. Untuk informasi selengkapnya, lihat [Dapatkan rekomendasi tipe instans untuk beban kerja baru](#).
- Lanjutan: Untuk menentukan atribut instans dan membiarkan Amazon EC2 mengidentifikasi tipe instans dengan atribut tersebut, pilih Lanjutan, lalu pilih Tentukan atribut tipe instans.
 - Jumlah vCPU: Masukkan jumlah minimum dan maksimum vCPU 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 [InstanceRequirementsMeminta](#) di Referensi API Amazon EC2.
 - 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)

Pasangan kunci untuk instans.

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](#).

Pengaturan jaringan

Konfigurasi pengaturan jaringan, sesuai keperluan.

- Subnet: Anda dapat meluncurkan sebuah instans di subnet yang terkait dengan Zona Ketersediaan, Local Zone, Wavelength Zone, atau Outpost.

Untuk meluncurkan instans di Zona Ketersediaan, pilih subnet tempat Anda akan meluncurkan instans. Untuk membuat subnet baru, pilih Buat subnet baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke wizard dan pilih ikon Segarkan untuk memuat subnet Anda dalam daftar.

Untuk meluncurkan instans di Local Zone, pilih subnet yang Anda buat di Local Zone.

Untuk meluncurkan sebuah instans di Outpost, pilih subnet di VPC yang Anda kaitkan dengan Outpost.

- Firewall (grup keamanan): Gunakan satu atau beberapa grup keamanan untuk menentukan aturan firewall untuk instans Anda. Aturan ini menentukan lalu lintas jaringan yang masuk yang dikirim ke instans Anda. Semua lalu lintas lainnya diabaikan. Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).

Jika Anda menambahkan antarmuka jaringan, Anda harus menentukan grup keamanan yang sama di antarmuka jaringan.

Pilih atau buat grup keamanan sebagai berikut:

- Untuk memilih grup keamanan yang ada, pilih Pilih grup keamanan yang ada, dan pilih grup keamanan Anda dari Grup keamanan umum.
- Untuk membuat grup keamanan baru, pilih Buat grup keamanan.

Anda dapat menambahkan aturan untuk menyesuaikan dengan kebutuhan Anda. Misalnya, jika instans Anda akan menjadi server web, buka port 80 (HTTP) dan 443 (HTTPS) untuk mengizinkan lalu lintas internet.

Untuk menambahkan aturan, pilih Tambahkan aturan grup keamanan. Untuk Tipe, pilih tipe lalu lintas jaringan. Bidang Protokol secara otomatis diisi dengan protokol untuk membuka lalu lintas jaringan. Untuk Tipe sumber, pilih tipe sumber. Untuk mengizinkan templat peluncuran menambahkan alamat IP publik komputer Anda, pilih IP Saya. Jika Anda terhubung melalui ISP atau dari belakang firewall Anda tanpa alamat IP statis, maka Anda harus menemukan rentang alamat IP yang digunakan oleh komputer klien.

⚠ Warning

Aturan yang mengaktifkan semua alamat IP ($0.0.0.0/0$) untuk mengakses instans Anda melalui SSH atau RDP dapat diterima jika Anda meluncurkan instans pengujian sebentar dan akan menghentikan atau mengakhirinya segera, tetapi tidak aman untuk lingkungan produksi. Anda hanya boleh mengotorisasi alamat IP atau rentang alamat tertentu saja untuk mengakses instans.

- Konfigurasi jaringan lanjutan

Antarmuka jaringan

- Indeks Perangkat: Nomor perangkat untuk antarmuka jaringan, misalnya `eth0`, untuk antarmuka jaringan utama. Jika Anda membiarkan bidang kosong, AWS akan membuat antarmuka jaringan utama.
- Antarmuka jaringan: Pilih Antarmuka baru agar Amazon EC2 dapat membuat antarmuka baru, atau memilih antarmuka jaringan yang ada dan tersedia.
- Deskripsi: (Opsional) Deskripsi untuk antarmuka jaringan baru.
- Subnet: Subnet tempat membuat antarmuka jaringan baru. Untuk antarmuka jaringan primer (`eth0`), ini adalah subnet tempat instans diluncurkan. Jika Anda telah memasukkan antarmuka jaringan yang ada untuk `eth0`, instans akan diluncurkan di subnet tempat antarmuka jaringan berada.
- Grup keamanan: Satu atau beberapa grup keamanan di VPC Anda yang akan digunakan untuk mengaitkan antarmuka jaringan.
- Tetapkan otomatis IP Publik: Tentukan apakah instans Anda menerima alamat IPv4 publik. Secara default, instans di subnet default menerima alamat IPv4 publik, sedangkan instans di subnet nondefault tidak menerimanya. Anda dapat memilih Aktifkan atau Nonaktifkan untuk mengganti pengaturan default subnet. Untuk informasi selengkapnya, lihat [Alamat IPv4 publik](#).
- IP Primer: Alamat IPv4 privat dari jangkauan rentang subnet Anda. Biarkan kosong agar Amazon EC2 dapat memilih alamat IPv4 privat untuk Anda.
- IP Sekunder: Satu atau beberapa alamat IPv4 privat tambahan dari jangkauan subnet Anda. Pilih Tetapkan secara manual dan masukkan alamat IP. Pilih Tambahkan IP untuk menambahkan alamat IP lain. Atau, pilih Tetapkan secara otomatis agar Amazon EC2 dapat memilih salah satu untuk Anda, dan masukkan nilai yang menunjukkan jumlah alamat IP yang akan ditambahkan.

- (IPv6 saja) IP IPv6: Alamat IPv6 dari rentang subnet. Pilih Tetapkan secara manual dan masukkan alamat IP. Pilih Tambahkan IP untuk menambahkan alamat IP lain. Atau, pilih Tetapkan secara otomatis agar Amazon EC2 dapat memilih salah satu untuk Anda, dan masukkan nilai yang menunjukkan jumlah alamat IP yang akan ditambahkan.
- Prefiks IPv4: Awalan IPv4 untuk antarmuka jaringan.
- Prefiks IPv6: Awalan IPv6 untuk antarmuka jaringan.
- (Opsional) Menetapkan IP IPv6 Primer: Jika Anda meluncurkan instans ke subnet tumpukan ganda atau IPv6 saja, Anda memiliki opsi untuk Menetapkan IP IPv6 Primer. Dengan menetapkan alamat IPv6 primer, Anda akan dapat menghindari mengganggu lalu lintas ke instans atau ENI. Pilih Aktifkan jika instans ini bergantung pada alamat IPv6 yang tidak berubah. Saat Anda meluncurkan instance, secara otomatis AWS akan menetapkan alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instans Anda menjadi alamat IPv6 utama. Setelah Anda mengaktifkan alamat IPv6 GUA 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 akan menjadi alamat IPv6 utama primer.
- Hapus saat pengakhiran: Apakah antarmuka jaringan akan dihapus saat instans dihapus.
- Elastic Fabric Adapter: Menunjukkan apakah antarmuka jaringan adalah Elastic Fabric Adapter. Untuk informasi selengkapnya, lihat [the section called “Elastic Fabric Adapter”](#).
- Indeks Kartu Jaringan: Indeks kartu jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0. Beberapa jenis RDS Support beberapa arus listrik 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. Template peluncuran tidak menyertakan konfigurasi ENA Ekspres untuk instans kecuali Anda memilih Aktifkan atau Nonaktifkan.
- ENA Express UDP: Jika Anda telah mengaktifkan ENA Ekspres, Anda dapat menggunakannya secara opsional untuk lalu lintas UDP. Template peluncuran tidak menyertakan konfigurasi ENA Ekspres untuk instans Anda kecuali Anda memilih Aktifkan atau Nonaktifkan.

Pilih Tambahkan antarmuka jaringan untuk menambahkan lebih banyak antarmuka jaringan. Jumlah antarmuka jaringan yang dapat Anda tambahkan tergantung pada nomor yang didukung

oleh tipe instans yang dipilih. Antarmuka jaringan tambahan dapat berada di subnet yang berbeda dari VPC yang sama atau di subnet di VPC berbeda yang Anda miliki (selama subnet berada di Availability Zone yang sama dengan instance Anda). Jika Anda memilih subnet di VPC lain, label multi-VPC muncul di sebelah antarmuka jaringan yang telah Anda tambahkan. Ini memungkinkan Anda membuat instans multi-homed di seluruh VPC dengan konfigurasi jaringan dan keamanan yang berbeda. Perhatikan bahwa jika Anda melampirkan ENI tambahan dari VPC lain, Anda harus memilih grup keamanan untuk ENI dari VPC tersebut.

Untuk informasi selengkapnya, lihat [Antarmuka jaringan elastis](#). Jika Anda menentukan lebih dari satu antarmuka jaringan, maka instans Anda tidak akan dapat menerima alamat IPv4 publik. Selain itu, jika Anda menentukan antarmuka jaringan yang ada untuk eth0, Anda tidak akan dapat mengganti pengaturan IPv4 publik subnet menggunakan Tetapkan Otomatis IP Publik. Untuk informasi selengkapnya, lihat [Menetapkan alamat IPv4 publik selama peluncuran instans](#).

Mengonfigurasi penyimpanan

Jika Anda menentukan AMI untuk templat peluncuran, AMI menyertakan satu atau lebih volume penyimpanan, termasuk volume root (Volume 1 (Root AMI)). 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).

Untuk menambahkan volume baru, pilih Tambahkan volume baru.

Dengan menggunakan tampilan Lanjutan, Anda dapat mengonfigurasi setiap volume sebagai berikut:

- Tipe penyimpanan: Tipe volume (EBS atau sementara) yang akan dikaitkan dengan instans Anda. Tipe volume penyimpanan instans (sementara) hanya tersedia jika Anda memilih tipe instans yang mendukungnya. Untuk informasi selengkapnya, lihat [Penyimpanan instans Amazon EC2](#) dan [volume Amazon EBS](#).
- Nama perangkat: Pilih dari daftar nama perangkat yang tersedia untuk volume.
- Snapshot: Pilih snapshot yang akan digunakan untuk membuat 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. Jika Anda telah memilih AMI dan instans yang memenuhi syarat untuk tingkat gratis, ingatlah bahwa agar tetap dalam tingkat gratis, Anda harus tetap di bawah 30 GiB dari total 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 tipe SSD IOPS yang tersedia (io1 dan io2) atau SSD Tujuan Umum (gp3), Anda dapat memasukkan jumlah operasi I/O per detik (IOPS) yang dapat didukung volume. Ini diperlukan untuk volume io1, io2, dan gp3. Ini tidak didukung untuk gp2, st1, sc1, atau volume standar. Jika Anda menghilangkan parameter 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.

Tanda sumber daya

Untuk memberi [tanda](#) pada sumber daya yang dibuat saat instans diluncurkan, di bawah Tanda sumber daya, pilih Tambahkan tanda, lalu masukkan kunci tanda 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
- Permintaan Instans Spot
- Antarmuka jaringan

Note

Untuk menandai templat peluncuran itu sendiri, Anda harus menentukan tag pada Tanda templat. Untuk informasi selengkapnya, lihat [Luncurkan nama template, deskripsi, dan tanda](#).

Detail lanjutan

Untuk Detail lanjutan, perluas bagian untuk melihat kolom dan menentukan parameter tambahan apa pun untuk instans.

- Opsi pembelian: Pilih Minta Instans Spot untuk meminta Instans Spot dengan harga Spot, yang tidak akan melebihi harga Sesuai Permintaan, dan pilih Sesuaikan 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, EC2 meluncurkan Instans Sesuai Permintaan secara default. Untuk informasi selengkapnya, lihat [Instans Spot](#).
- Profil instans IAM: Pilih profil instans (IAM) AWS Identity and Access Management untuk dikaitkan dengan instans. 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 [Tipe nama host instans Amazon EC2](#).
- Nama Host DNS: Menentukan apakah permintaan DNS ke nama sumber daya atau nama IP (tergantung pada pilihan Anda untuk Tipe hostname) akan merespons dengan alamat IPv4 (catatan A), alamat IPv6 (catatan AAAA), atau keduanya. Untuk informasi selengkapnya, lihat [Tipe nama host instans Amazon EC2](#).
- Perilaku pematian: Pilih apakah instans harus berhenti atau diakhiri saat dimatikan. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh instans](#).
- Berhenti - Perilaku hibernasi: Untuk mengaktifkan hibernasi, pilih Aktifkan. Bidang ini hanya valid untuk instans yang 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 [Aktifkan perlindungan pengakhiran](#).
- 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 menggunakan Amazon CloudWatch. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat [Pantau instans Anda menggunakan CloudWatch](#).
- GPU Elastis: Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.
- Inferensi Elastis: Akselerator inferensi elastis untuk dipasang ke instans CPU EC2 Anda. Untuk informasi selengkapnya, lihat [Bekerja dengan Amazon Elastic Inference](#) dalam Panduan Developer Amazon Elastic Inference.

Note

Mulai 15 April 2023, tidak AWS akan memasukkan pelanggan baru ke Amazon Elastic Inference (EI), dan akan membantu pelanggan saat ini memigrasikan beban kerja mereka ke opsi yang menawarkan harga dan kinerja yang lebih baik. Setelah 15 April 2023, pelanggan baru tidak akan dapat meluncurkan instans dengan akselerator Amazon EI di Amazon, Amazon ECS, atau SageMaker Amazon EC2. Namun, pelanggan yang telah menggunakan Amazon EI setidaknya sekali selama periode 30 hari terakhir dianggap sebagai pelanggan saat ini dan akan dapat terus menggunakan layanan ini.

- 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 [Instans performa yang dapat melonjak](#).
- Nama grup penempatan: Tentukan grup penempatan untuk meluncurkan instans. Anda dapat memilih grup penempatan yang sudah ada, atau membuat grup yang baru. Tidak semua tipe instans dapat diluncurkan dalam grup penempatan. Untuk informasi selengkapnya, lihat [Grup penempatan](#).
- Instans dengan pengotimalan EBS: Pilih Aktifkan untuk menyediakan kapasitas tambahan khusus untuk I/O Amazon EBS. Tidak semua tipe instans mendukung fitur ini. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat [the section called "Optimisasi EBS"](#).
- 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 lebih lanjut, lihat [Instans Khusus](#) dan [Host Khusus](#).
- **ID disk RAM:** (Hanya berlaku untuk AMI paravirtual (PV)) Pilih disk RAM untuk instans. Jika Anda telah memilih kernel, Anda mungkin perlu memilih RAM disk tertentu dengan driver untuk mendukungnya.
- **Id Kernel:** (Hanya berlaku untuk AMI paravirtual (PV)) Pilih kernel untuk instans.
- **Nitro Enclave:** Memungkinkan Anda untuk membuat lingkungan eksekusi terisolasi, yang disebut enclaves, dari instans Amazon EC2. Pilih Aktifkan untuk mengaktifkan instans untuk AWS Nitro Enclave. Untuk informasi selengkapnya, lihat [Apa itu AWS Nitro Enclaves?](#) di Panduan Pengguna AWS Nitro Enclaves.
- **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:** Pilih Tentukan opsi CPU untuk menentukan jumlah vCPU kustom vCPUs selama peluncuran. Atur jumlah inti CPU dan thread per inti. Untuk informasi selengkapnya, lihat [Mengoptimalkan opsi CPU](#).
- **Titik akhir IPv6 metadata:** Anda dapat mengaktifkan instance untuk menggunakan alamat IPv6 IMDS untuk mengambil metadata instance. [fd00:ec2::254] Opsi ini hanya tersedia jika Anda meluncurkan [instance yang dibangun di atas Sistem AWS Nitro](#) ke dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#). Untuk informasi selengkapnya, lihat [Mengambil metadata instans](#).
- **Metadata dapat diakses:** Anda dapat mengaktifkan atau menonaktifkan akses ke IMDS. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans untuk instans baru](#).
- **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 instansnya dari metadatanya. Jika Anda tidak menyertakan pengaturan ini dalam

templat, secara default, akses ke tanda dalam metadata instans tidak diperbolehkan. Untuk informasi selengkapnya, lihat [Mengizinkan akses ke tanda dalam metadata instans](#).

- Data pengguna: Anda dapat menentukan data pengguna untuk mengonfigurasi instans selama peluncuran, atau untuk menjalankan skrip konfigurasi. Untuk informasi selengkapnya, lihat [Jalankan perintah pada instans Amazon EC2 Anda saat peluncuran](#).

AWS CLI contoh

Contoh berikut menggunakan perintah [create-launch-template untuk membuat template](#) peluncuran dengan nama dan konfigurasi instance yang ditentukan.

```
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-7b16de0c"  
  }],  
  "ImageId": "ami-8c1be5f6",  
  "InstanceType": "r4.4xlarge",  
  "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"
  }
}
```

AWS Tools for Windows PowerShell contoh

Contoh berikut menggunakan [New-EC2LaunchTemplate](#) cmdlet untuk membuat template peluncuran dengan nama dan konfigurasi instance yang ditentukan.

```
$launchTemplateData = [Amazon.EC2.Model.RequestLaunchTemplateData]@{
  ImageId = 'ami-8c1be5f6'
  InstanceType = 'r4.4xlarge'
  NetworkInterfaces = @(
    [Amazon.EC2.Model.LaunchTemplateInstanceNetworkInterfaceSpecificationRequest]@{
      AssociatePublicIpAddress = $true
      DeviceIndex = 0
      Ipv6AddressCount = 1
      SubnetId = 'subnet-7b16de0c'
    }
  )
  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.

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 memberi tanda pada templat peluncuran saat pembuatan, perluas Tanda templat, pilih Tambahkan tanda, lalu masukkan kunci tanda 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

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 Contoh.
3. Pilih instans, dan pilih Tindakan, Buat templat dari instans.
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, dan kemudian 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": "2600:0000:0000:0000:0000:0000:0000:0000"  
        }  
      ]  
    }  
  ]  
}
```

```

        {
            "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, lihat [Buat template peluncuran dari parameter](#).

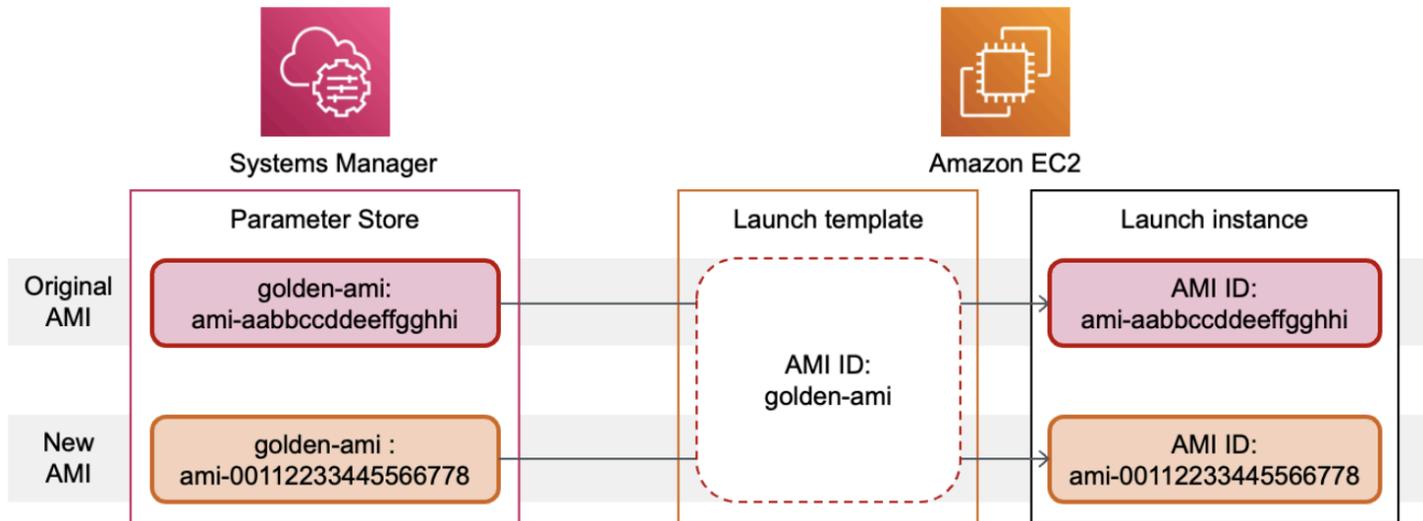
Gunakan parameter Systems Manager alih-alih ID AMI

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 dibagikan 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 Systems Manager adalah pasangan nilai-kunci yang ditentukan pengguna yang Anda buat di Systems Manager Parameter Store. Parameter Store menyediakan penyimpanan pusat untuk menyimpan nilai konfigurasi aplikasi Anda. Untuk informasi selengkapnya, lihat [Penyimpanan Parameter AWS Systems Manager](#) dalam Panduan Pengguna AWS Systems Manager.

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 Tentukan berdasarkan nilai kustom/parameter Systems Manager.
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 dari 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 Systems Manager ke ID AMI yang sebenarnya

Gunakan [perintah describe-launch-template-versions](#) dan sertakan parameternya. `--resolve-alias`

```
aws ec2 describe-launch-template-versions \
--launch-template-name my-launch-template \
--versions $Default \
--resolve-alias
```

Tanggapan tersebut mencakup ID AMI untuk ImageId. Dalam contoh ini, ketika sebuah instance diluncurkan menggunakan template peluncuran ini, ID AMI akan menyelesaikannya. `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 Systems Manager, lihat materi referensi berikut dalam dokumentasi Systems Manager.

- Untuk informasi tentang cara mencari parameter publik AMI yang didukung oleh Amazon EC2, lihat Memanggil parameter [publik AMI](#).
- Untuk informasi tentang berbagi parameter dengan AWS akun lain atau melalui AWS Organizations, lihat [Bekerja dengan parameter bersama](#).
- Untuk informasi tentang pemantauan apakah parameter berhasil dibuat, lihat [Dukungan parameter asli untuk ID Gambar Mesin Amazon](#).

Batasan

- Saat ini, Armada EC2 dan Armada Spot tidak mendukung penggunaan template peluncuran yang menentukan parameter Systems Manager sebagai pengganti ID AMI. Untuk Armada EC2 dan Armada Spot, jika Anda menentukan AMI di templat peluncuran, Anda harus menentukan ID AMI.
- Auto Scaling Amazon EC2 memberikan batasan lain. Untuk informasi selengkapnya, lihat [Menggunakan AWS Systems Manager parameter alih-alih ID AMI di templat peluncuran](#) di Panduan Pengguna Auto Scaling 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 templat peluncuran versi lain, mengatur versi default, menjelaskan versi templat peluncuran, dan menghapus versi yang tidak lagi Anda perlukan.

Tugas

- [Buat versi templat peluncuran](#)
- [Menyetel versi templat peluncuran default](#)
- [Jelaskan versi templat peluncuran](#)
- [Hapus 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 informasi selengkapnya tentang parameter peluncuran, lihat [Membuat templat peluncuran](#).

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 templat peluncuran, lalu pilih Tindakan, Ubah templat (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. Modifikasi parameter peluncuran sesuai kebutuhan, dan 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-c998b6b2"
```

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 templat peluncuran dan pilih Tindakan, Setel 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](#) dan tentukan versi yang ingin Anda atur sebagai default.

```
aws ec2 modify-launch-template \  
  --launch-template-id lt-0abcd290751193123 \  
  --default-version 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 rentang versi templat peluncuran yang ditentukan. 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 templat peluncuran: Pilih templat 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 semua versi templat 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"
```

Hapus versi templat peluncuran

Jika Anda tidak lagi memerlukan versi templat peluncuran, Anda dapat menghapusnya.

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 templat peluncuran dan pilih Tindakan, Hapus versi templat.
4. Pilih versi yang akan dihapus lalu pilih Hapus.

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
```

Hapus templat peluncuran

Jika Anda tidak lagi memerlukan templat peluncuran, Anda dapat menghapusnya. Menghapus templat peluncuran akan menghapus semua versinya. Untuk menghapus versi templat peluncuran tertentu, lihat [Hapus versi templat peluncuran](#).

Saat Anda menghapus templat peluncuran, hal ini tidak memengaruhi instans apa pun yang telah Anda luncurkan dari templat peluncuran.

Console

Untuk menghapus templat peluncuran

- Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
- Di panel navigasi, pilih Templat Peluncuran.
- Pilih templat peluncuran dan pilih Tindakan, Hapus templat.
- Masukkan **Delete** untuk mengonfirmasi penghapusan, lalu pilih Hapus.

AWS CLI

Untuk menghapus templat peluncuran

- Gunakan perintah (AWS CLI) [delete-launch-template](#) dan tentukan templat peluncurannya.

```
aws ec2 delete-launch-template --launch-template-id lt-01238c059e3466abc
```

Meluncurkan instans dari templat peluncuran

Templat peluncuran didukung oleh beberapa layanan peluncuran instans. Topik ini menjelaskan cara menggunakan templat peluncuran saat meluncurkan instans menggunakan wizard peluncuran instans EC2, Amazon EC2 Auto Scaling, Armada EC2, dan Armada Spot.

Topik

- [Meluncurkan sebuah instans dari templat peluncuran](#)
- [Gunakan templat peluncuran dengan Amazon EC2 Auto Scaling](#)
- [Gunakan templat peluncuran dengan Armada EC2](#)
- [Gunakan templat peluncuran dengan Armada Spot](#)

Meluncurkan sebuah instans dari templat peluncuran

Anda dapat menggunakan parameter yang terdapat dalam templat peluncuran untuk meluncurkan sebuah instans. Anda memiliki opsi untuk mengganti atau menambahkan parameter peluncuran sebelum Anda meluncurkan instans.

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 tanda ini.

Console

Untuk meluncurkan sebuah instans dari template peluncuran menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Pilih templat peluncuran dan pilih Tindakan, Luncurkan instans dari templat.
4. Untuk Versi templat sumber, pilih versi templat peluncuran yang akan digunakan.
5. Untuk Jumlah instans, tentukan jumlah instans yang akan diluncurkan.
6. (Opsional) Anda dapat mengganti atau menambahkan parameter templat peluncuran dengan mengubah dan menambahkan parameter di bagian Detail instans.
7. Pilih Luncurkan instans dari templat.

AWS CLI

Untuk meluncurkan sebuah instans dari templat peluncuran menggunakan AWS CLI

- 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 [Pemecahan masalah peluncuran instans](#).

PowerShell

Untuk meluncurkan sebuah instans dari templat peluncuran menggunakan AWS Tools for PowerShell

- Gunakan [New-EC2Instance](#) perintah dan tentukan `-LaunchTemplate` parameternya. 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 [Pemecahan masalah peluncuran instans](#).

Gunakan templat peluncuran dengan Amazon EC2 Auto Scaling

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. Untuk informasi selengkapnya, lihat [Membuat template peluncuran untuk grup Auto Scaling dan Membuat template peluncuran menggunakan setelan lanjutan](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Sebelum Anda dapat membuat grup Auto Scaling menggunakan templat peluncuran, Anda harus membuat templat peluncuran yang menyertakan parameter yang diperlukan untuk meluncurkan sebuah instans dalam grup Auto Scaling, seperti ID AMI. Konsol menyediakan panduan untuk membantu Anda membuat template yang dapat Anda gunakan dengan Auto Scaling Amazon EC2.

Untuk membuat templat peluncuran yang akan digunakan dengan Auto Scaling menggunakan konsol

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. Di bawah panduan Auto Scaling, pilih kotak centang agar Amazon EC2 memberikan panduan untuk membantu membuat templat untuk digunakan dengan Auto Scaling.
6. Ubah parameter peluncuran sesuai kebutuhan. Karena Anda memilih panduan Auto Scaling, beberapa bidang wajib diisi dan beberapa bidang tidak tersedia. Untuk informasi tentang cara

mengonfigurasi parameter peluncuran untuk Auto Scaling Amazon EC2, [lihat Membuat templat peluncuran untuk grup Auto Scaling dan Membuat templat peluncuran menggunakan setelan lanjutan di Panduan Pengguna Auto Scaling](#) Amazon EC2.

7. Pilih Buat templat peluncuran.
8. (Opsional) Untuk membuat grup Auto Scaling menggunakan templat peluncuran ini, di halaman Langkah berikutnya, pilih Buat grup Auto Scaling.

Untuk contoh yang menunjukkan cara menggunakan templat peluncuran dengan berbagai kombinasi parameter, lihat [Contoh untuk membuat dan mengelola templat peluncuran dengan AWS Command Line Interface \(AWS CLI\)](#) di Panduan Pengguna Auto Scaling Amazon EC2. AWS CLI

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 tentang membuat atau memperbarui grup Auto Scaling menggunakan templat peluncuran, lihat topik berikut di Panduan Pengguna Auto Scaling Amazon EC2.

- [Buat grup Auto Scaling menggunakan template peluncuran](#)
- [Memperbarui grup Auto Scaling](#)

Gunakan templat peluncuran dengan Armada EC2

Anda dapat membuat permintaan Armada EC2 dan menentukan templat peluncuran dalam konfigurasi instans. 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.

Gunakan templat peluncuran dengan Armada Spot

Anda dapat membuat permintaan Armada Spot dan menentukan templat peluncuran dalam konfigurasi instans. Saat Amazon EC2 memenuhi permintaan Armada Spot, 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 permintaan Armada Spot](#).

Untuk membuat permintaan Armada Spot dengan templat peluncuran menggunakan konsol

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. Konfigurasi Armada Spot Anda dengan memilih opsi yang berbeda di layar ini. Untuk informasi lebih lanjut tentang opsi, lihat [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).
7. Saat Anda siap membuat Armada Spot, pilih Luncurkan.

Untuk membuat permintaan Spot Fleet dengan template peluncuran menggunakan AWS CLI

- Gunakan perintah [request-spot-fleet](#). Gunakan parameter `LaunchTemplateConfigs` untuk menentukan templat peluncuran dan setiap penggantian untuk templat peluncuran.

Meluncurkan sebuah instans menggunakan parameter dari instans yang ada

Konsol Amazon EC2 menyediakan opsi Luncurkan 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

- Kami tidak mengklonkan 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, maka instans baru akan 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)
- Pengaturan pengoptimalan Amazon EBS (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
- [Contoh 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 penggabungan domain, lihat [Bergabung dengan Instans EC2 Windows dengan lancar](#) di Panduan Administrasi AWS Directory Service .

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 [Pemecahan masalah peluncuran instans](#).

Luncurkan sebuah AWS Marketplace instance

Anda dapat berlangganan AWS Marketplace produk dan meluncurkan instance dari AMI produk menggunakan wizard peluncuran Amazon EC2. Untuk informasi lebih lanjut tentang AMI berbayar, lihat [AMI berbayar](#). Untuk membatalkan langganan Anda setelah peluncuran, Anda harus terlebih dahulu mengakhiri semua instans yang berjalan darinya. Untuk informasi selengkapnya, lihat [Kelola AWS Marketplace langganan Anda](#).

New console

Untuk meluncurkan instance dari AWS Marketplace menggunakan wizard peluncuran

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. Pada Aplikasi dan Gambar OS (Amazon Machine Image), pilih Jelajahi AMI lainnya, lalu pilih tab AWS Marketplace AMI. 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 salah satu tombol berikut:
 - Berlangganan saat peluncuran instans — Langganan Anda dimulai saat Anda memilih Launch instance (pada Langkah 10).
 - Berlangganan sekarang — Langganan Anda segera dimulai. 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, Firewall (grup keamanan), perhatikan grup keamanan baru yang dibuat sesuai dengan spesifikasi vendor untuk produk tersebut. Grup keamanan mungkin menyertakan aturan yang mengizinkan semua akses alamat IPv4 (0.0.0.0/0) pada SSH (port 22) di Linux atau RDP (port 3389) di 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 konfigurasi, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).

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 Anda memilih Berlangganan saat peluncuran instans pada Langkah 5, Anda terlebih dahulu berlangganan produk sebelum instans Anda dapat 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 [Hubungkan ke instans Linux Anda](#) dan [Hubungkan ke instans Windows Anda](#).

 Important

Periksa instruksi 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 AWS Marketplace langganan Anda](#).

13. Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat [Pemecahan masalah peluncuran instans](#).

Old console

Untuk meluncurkan instance dari AWS Marketplace menggunakan wizard peluncuran

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari dasbor Amazon EC2, pilih Luncurkan Instans.

3. Di halaman Pilih Amazon Machine Image (AMI), pilih kategori AWS Marketplace di sebelah kiri. Temukan AMI yang sesuai dengan menelusuri kategori, atau menggunakan fungsi pencarian. Pilih Pilih untuk memilih produk Anda.
4. Dialog menampilkan gambaran umum produk yang Anda pilih. Anda dapat melihat informasi harga, serta informasi lain yang disediakan vendor. Saat Anda siap, pilih Lanjutkan.

 Note

Anda tidak dikenai biaya untuk penggunaan produk hingga Anda meluncurkan instans dengan AMI. Catat harga untuk setiap tipe instans yang didukung, karena Anda akan diminta untuk memilih tipe instans di halaman wizard berikutnya. Pajak tambahan mungkin juga berlaku pada produk.

5. Di halaman Pilih Tipe Instans, pilih konfigurasi perangkat keras dan ukuran instans yang akan diluncurkan. Setelah selesai, pilih Berikutnya: Konfigurasikan Detail Instans.
6. Di halaman wizard berikutnya, Anda dapat mengonfigurasi instans Anda, menambah penyimpanan, dan menambahkan tanda. Untuk informasi selengkapnya tentang berbagai opsi yang dapat Anda konfigurasi, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#). Pilih Berikutnya hingga Anda mencapai halaman Konfigurasi Grup Keamanan.

Wizard membuat grup keamanan baru sesuai dengan spesifikasi vendor untuk produk tersebut. Grup keamanan mungkin menyertakan aturan yang mengizinkan semua akses alamat IPv4 (0.0.0.0/0) pada SSH (port 22) di Linux atau RDP (port 3389) di Windows. Kami menyarankan Anda menyesuaikan aturan ini untuk mengizinkan hanya alamat atau rentang alamat tertentu yang bisa mengakses instans Anda melalui port tersebut.

Saat Anda siap, pilih Tinjau dan Luncurkan.

7. Pada halaman Tinjau Peluncuran Instans, periksa detail AMI tempat Anda akan meluncurkan instans, serta detail konfigurasi lain yang Anda siapkan di wizard. Saat Anda siap, pilih Luncurkan untuk memilih atau membuat pasangan kunci, dan luncurkan instans Anda.
8. Tergantung pada produk langganannya, instans mungkin memerlukan waktu beberapa menit atau lebih untuk diluncurkan. Anda terlebih dahulu berlangganan produk sebelum instans Anda dapat diluncurkan. Jika ada masalah dengan detail kartu kredit Anda, Anda akan diminta untuk memperbarui detail akun Anda. Saat halaman konfirmasi peluncuran ditampilkan, pilih Lihat instans untuk membuka halaman Instans.

Note

Anda akan dikenakan harga langganan selama instans Anda berjalan, meskipun sedang idle. Jika instans Anda dihentikan, Anda mungkin masih dikenakan biaya untuk penyimpanan.

9. Saat instans Anda ada dalam status `running`, Anda dapat menyambungkannya. Untuk melakukan ini, pilih instans Anda di daftar dan pilih **Hubungkan**. Ikuti instruksi di dialog. Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat [Hubungkan ke instans Linux Anda](#) dan [Hubungkan ke instans Windows Anda](#).

Important

Periksa instruksi penggunaan vendor dengan hati-hati, karena Anda mungkin perlu menggunakan nama pengguna tertentu untuk masuk ke instans. Untuk informasi lebih lanjut tentang mengakses rincian langganan Anda, lihat [Kelola AWS Marketplace langganan Anda](#).

10. Jika instance gagal diluncurkan atau status langsung beralih ke `terminated` dari pada `running`, lihat [Pemecahan masalah peluncuran instans](#).

Luncurkan instance AWS Marketplace AMI menggunakan API dan CLI

Untuk meluncurkan instance dari AWS Marketplace produk yang menggunakan API atau 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-instances , atau lihat topik berikut ini untuk informasi selengkapnya: Meluncurkan Instans .
AWS Tools for Windows PowerShell	Gunakan New-EC2Instance perintah, atau lihat topik berikut untuk informasi selengkapnya: Luncurkan Instans Amazon EC2 Menggunakan Windows PowerShell
API Kueri	Menggunakan RunInstances permintaan.

Hentikan dan mulai instans Amazon EC2

Anda dapat menghentikan dan memulai instans Anda jika instans memiliki volume Amazon EBS sebagai perangkat root-nya. Ketika Anda menghentikan sebuah instance, itu mati. Ketika Anda memulai sebuah instance, biasanya dimigrasikan ke komputer host baru yang mendasarinya dan diberi alamat IPv4 publik baru.

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 [Mengakhiri 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 boot ulang, penghentian, hibernasi, dan pengakhiran](#).

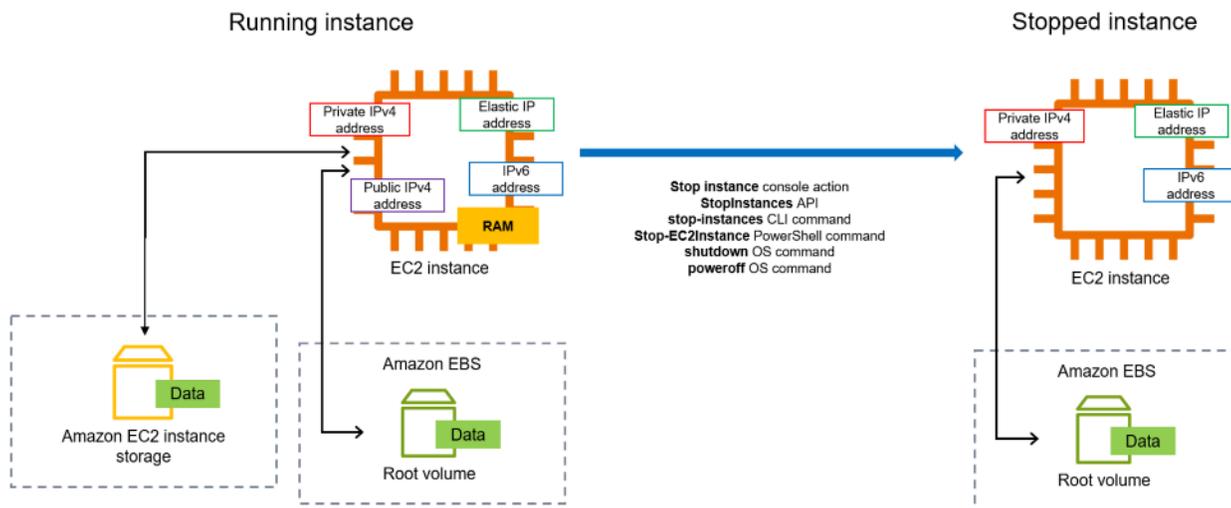
Daftar Isi

- [Bagaimana instans berhenti dan mulai bekerja](#)
- [Hentikan dan mulai instans Anda secara manual](#)
- [Menghentikan dan memulai instans Anda secara otomatis](#)
- [Temukan semua instans yang berjalan dan berhenti](#)
- [Aktifkan perlindungan berhenti untuk instans Anda](#)

Bagaimana instans berhenti dan mulai bekerja

Ketika Anda menghentikan sebuah instance, perubahan terdaftar pada tingkat OS instance, beberapa sumber daya hilang, dan beberapa sumber daya tetap ada. Saat Anda memulai sebuah instans, perubahan terdaftar di tingkat instans.

Diagram berikut menunjukkan apa yang hilang dan apa yang bertahan ketika instans Amazon EC2 dihentikan. Ketika sebuah instans berhenti, instans kehilangan volume penyimpanan instans terlampir dan data yang disimpan pada volume tersebut, data yang disimpan pada RAM instans, dan alamat IPv4 publik yang ditetapkan, jika alamat IP Elastis tidak terkait dengan instans. Sebuah instans mempertahankan alamat IPv4 privat yang ditetapkan, alamat IP Elastis yang terkait dengan instans, alamat IPv6 apa pun, dan setiap volume Amazon EBS yang terlampir serta data pada volume tersebut.



Apa yang terjadi jika Anda menghentikan 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. Pemastian yang tertib dipicu oleh peristiwa penekanan tombol pemastian ACPI dari hypervisor.
- Pemastian ACPI dimulai.
- Instans dimatikan saat proses pemastian terkontrol keluar. Tidak ada waktu pemastian OS yang dapat dikonfigurasi.
- Jika OS instans tidak dimatikan dengan bersih dalam beberapa menit, pemastian keras dilakukan.
- Instans tersebut berhenti berjalan.
- Status instans berubah menjadi `stopping` kemudian `stopped`.
- [Penskalaan Otomatis] Jika instans Anda berada dalam grup Auto Scaling, saat instans berada dalam status Amazon EC2 selain `running`, atau jika statusnya untuk pemeriksaan status menjadi `impaired`, Amazon EC2 Auto Scaling menganggap instans tersebut tidak sehat dan menggantikannya. Untuk informasi selengkapnya, lihat [Pemeriksaan kondisi untuk instans Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- [Instans Windows] Saat Anda menghentikan dan memulai instance Windows, agen peluncuran melakukan tugas pada instance, seperti mengubah huruf drive untuk volume Amazon EBS yang terlampir. Untuk informasi selengkapnya tentang default ini dan bagaimana Anda dapat mengubahnya, lihat [the 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

- Setiap volume Amazon EBS yang terlampir.
- Data yang disimpan pada volume Amazon EBS terlampir.
- 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](#).

Untuk informasi tentang apa yang terjadi ketika Anda menghentikan instance Mac, lihat [the section called “Menghentikan dan mengakhiri instans Mac Anda”](#).

Apa yang terjadi jika Anda memulai sebuah instans

Perubahan terdaftar di tingkat OS

- 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](#)).
- Amazon EC2 menetapkan alamat IPv4 publik baru ke instans jika instans dikonfigurasi untuk menerima alamat IPv4 publik. Untuk mempertahankan alamat IPv4 publik yang tidak pernah berubah, Anda dapat mengaitkan [alamat IP Elastis](#) dengan instans Anda.

Uji respons aplikasi untuk berhenti dan mulai

Anda dapat menggunakan AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons ketika instance Anda dihentikan dan dimulai. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS Fault Injection Service](#).

Biaya yang terkait dengan instance stop and start

Biaya berikut dikaitkan dengan menghentikan dan memulai sebuah instans.

Berhenti—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 dikenai 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.

Hentikan dan mulai instans Anda secara manual

Anda dapat menghentikan dan memulai instans yang didukung Amazon EBS (instans dengan perangkat root EBS). Anda tidak dapat berhenti dan memulai instance dengan perangkat root penyimpanan instance.

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 perlukan dari volume penyimpanan instans ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.

Console

Untuk menghentikan dan memulai instans yang didukung Amazon EBS

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Instans, lalu pilih instance.
3. Pada tab Penyimpanan, verifikasi bahwa jenis perangkat Root adalah EBS. Jika tidak, Anda tidak dapat menghentikan instance.
4. Pilih Status instans, Hentikan instans. Jika opsi ini dinonaktifkan, baik instans sudah dihentikan maupun perangkat root-nya adalah volume penyimpanan instans.

5. Ketika diminta konfirmasi, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
6. Untuk memulai instans yang berhenti, pilih instans, dan pilih Status instans, Mulai instans.
7. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`.
8. Jika Anda menghentikan instans yang didukung Amazon EBS dan instans tersebut tampak “macet” di status `stopping`, Anda dapat menghentikannya secara paksa. Untuk informasi selengkapnya, lihat [Pemecahan masalah penghentian instans Anda](#).

Command line

Prasyarat

Verifikasi bahwa perangkat root instance adalah volume EBS. Misalnya, jalankan AWS CLI perintah [describe-instance](#) dan verifikasi `ituRootDeviceType`, bukan `ebs instance-store`

Untuk menghentikan dan memulai instans yang didukung Amazon EBS

Gunakan salah satu perintah berikut:

- AWS CLI—[stop-instances](#) dan [start-instances](#).
- AWS Tools for PowerShell— [Stop-EC2Instance](#) dan [Start-EC2Instance](#).
- Perintah OS—Anda dapat menginisiasi pematian menggunakan perintah `shutdown` atau `poweroff`. Saat Anda menggunakan perintah OS, instans berhenti secara default. Anda dapat mengubah perilaku ini sehingga berakhir. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh instans](#).

[Instance Linux] Menggunakan `halt` perintah OS dari sebuah 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.

Menghentikan dan memulai instans Anda secara otomatis

Anda dapat mengotomatisasi penghentian dan pemulaian instans dengan layanan berikut:

Penjadwal Instance aktif AWS

Anda dapat menggunakan Penjadwal Instance aktif AWS untuk mengotomatiskan awal dan penghentian instans EC2. Untuk informasi selengkapnya, lihat [Bagaimana cara menggunakan](#)

[Penjadwal Instance untuk CloudFormation 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 Anda 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 berjalan dan berhenti di semua Wilayah AWS pada 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 [Amazon EC2 Global View](#).

Aktifkan perlindungan berhenti untuk instans 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.

DisableApiStopAtribut [ModifyInstanceAttribute](#) API Amazon EC2 mengontrol apakah instans 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

- Mengaktifkan perlindungan penghentian tidak menghindarkan Anda dari penghentian instans secara tidak sengaja dengan memulai pematikan dari instans menggunakan perintah sistem operasi seperti shutdown atau poweroff.

- Mengaktifkan perlindungan berhenti tidak AWS mencegah menghentikan instance ketika ada [acara terjadwal](#) untuk menghentikan instance.
- 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](#).
- Stop protection 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 ke `false`, setelah `DisableApiTermination` atribut menentukan apakah instance dapat dihentikan menggunakan konsol, AWS CLI, atau API. Untuk mengetahui informasi selengkapnya, lihat [Mengakhiri instans Amazon EC2](#).
- Anda tidak dapat mengaktifkan perlindungan penghentian untuk instans yang didukung 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 penghentian untuk suatu instans saat meluncurkan instans menggunakan salah satu metode berikut ini.

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. Konfigurasi 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 AWS CLI perintah [run-instance](#) untuk meluncurkan instance, dan tentukan parameternya. `disable-api-stop`

```
aws ec2 run-instances \  
  --image-id ami-a1b2c3d4e5example \  
  --instance-type t3.micro \  
  --key-name MyKeyPair \  
  --disable-api-stop \  
  ...
```

Aktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

Anda dapat mengaktifkan perlindungan penghentian untuk suatu instans saat instans sedang berjalan atau berhenti menggunakan metode berikut ini.

Console

Untuk mengaktifkan 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 Tindakan > Pengaturan instans > Ubah perlindungan penghentian.
4. Pilih kotak centang Aktifkan, lalu pilih Simpan.

AWS CLI

Untuk mengaktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

Gunakan perintah [modify-instance-attribute](#) AWS CLI dan tentukan parameternya. `disable-api-stop`

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --disable-api-stop
```

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 Simpan.

AWS CLI

Untuk menonaktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

Gunakan perintah [modify-instance-attribute](#) AWS CLI dan tentukan parameternya. `no-disable-api-stop`

```
aws ec2 modify-instance-attribute \  
  --instance-id i-1234567890abcdef0 \  
  --no-disable-api-stop
```

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 instans, 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.

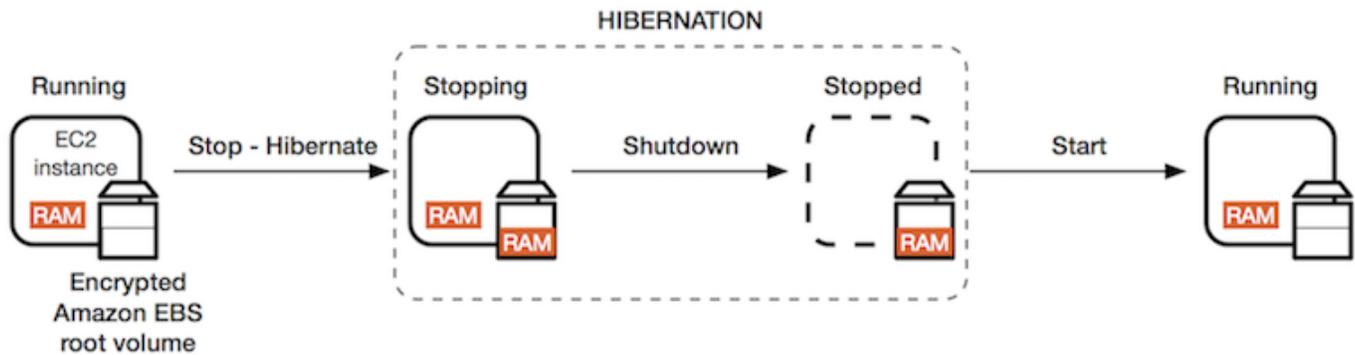
Jika Anda tidak lagi membutuhkan sebuah instans, Anda dapat mengakhirinya kapan saja, termasuk saat berada dalam status `stopped` (hibernasi). Untuk informasi selengkapnya, lihat [Mengakhiri instans Amazon EC2](#).

Daftar Isi

- [Cara kerja hibernasi instans Amazon EC2](#)
- [Prasyarat untuk hibernasi instans Amazon EC2](#)
- [Konfigurasi 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 gambaran dasar dari proses hibernasi untuk instans EC2.



Apa yang terjadi ketika Anda hibernasi sebuah instance

Saat Anda hibernasi sebuah instance, hal berikut terjadi:

- Instance 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 terlampir pada instans, tetapi data pada volume penyimpanan instans hilang.
- Saat instans Anda ada dalam status `stopped`, Anda dapat memodifikasi atribut tertentu dari instans, termasuk tipe atau ukuran instans.
- Dalam kebanyakan kasus, instans 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 boot ulang, penghentian, hibernasi, dan pengakhiran](#).

Batasan

- Ketika Anda menghibernasi suatu instans, data pada setiap volume penyimpanan instans akan hilang.
- (Instance Linux) Anda tidak dapat hibernasi instance Linux yang memiliki lebih dari 150 GB RAM.
- (Instans Windows) Anda tidak dapat hibernasi instance Windows yang memiliki lebih dari 16 GB RAM.
- 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 [Health memeriksa instans di grup Auto Scaling](#) di Panduan Pengguna Auto Scaling Amazon EC2.
- Anda tidak dapat hibernasi instance yang dikonfigurasi untuk boot dalam mode UEFI dengan [UEFI Secure Boot](#) 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](#).

- Ada beberapa cara di mana Anda dapat mengaktifkan hibernasi untuk Instans Spot. Untuk informasi selengkapnya, lihat [Menentukan perilaku interupsi](#).

Prasyarat untuk hibernasi instans Amazon EC2

Anda dapat mengaktifkan dukungan hibernasi untuk Instans Sesuai Permintaan atau Instans Spot saat meluncurkannya. Anda tidak dapat mengaktifkan hibernasi pada instance yang ada, baik sedang berjalan atau dihentikan. Untuk informasi selengkapnya, lihat [Aktifkan hibernasi instance](#).

Persyaratan untuk hibernasi instance

- [Wilayah AWS](#)
- [AMI](#)
- [Keluarga contoh](#)
- [Ukuran RAM instans](#)
- [Tipe volume root](#)

- [Ukuran volume akar](#)
- [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 AL2023 yang dirilis pada 20/09/2023 atau setelahnya
- AMI Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya
- AMI Amazon Linux 2018.03 yang dirilis 16.11.2018 atau setelahnya
- AMI CentOS versi 8 ¹ ([Konfigurasi tambahan](#) diperlukan)
- AMI Fedora versi 34 atau setelahnya ¹ ([Konfigurasi tambahan](#) diperlukan)
- AMI Red Hat Enterprise Linux (RHEL) 9 ¹ ([Konfigurasi tambahan](#) diperlukan)
- AMI Red Hat Enterprise Linux (RHEL) 8 ¹ ([Konfigurasi tambahan](#) diperlukan)
- AMI Ubuntu 22.04.2 LTS (Jammy Jellyfish) dirilis dengan nomor seri 20230303 atau setelahnya ²
- AMI Ubuntu 20.04 LTS (Focal Fossa) dirilis dengan nomor seri 20210820 atau setelahnya ²
- AMI Ubuntu 18.04 LTS (Bionic Beaver) dirilis dengan nomor seri 20190722.1 atau setelahnya ^{2 4}
- AMI Ubuntu 16.04 LTS (Xenial Xerus) ^{2 3 4} ([Konfigurasi tambahan](#) diperlukan)

¹ Untuk CentOS, Fedora, dan Red Hat Enterprise Linux, hibernasi hanya didukung pada instans berbasis Nitro.

² Kami merekomendasikan untuk menonaktifkan KASLR pada instans 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 \(khusus Ubuntu\)](#).

³ Untuk AMI Ubuntu 16.04 LTS (Xenial Xerus), hibernasi tidak didukung pada tipe 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).

Dukungan untuk Ubuntu 18.04 LTS (Bionic Beaver) dan Ubuntu 16.04 LTS (Xenial Xerus) telah mencapai akhir masa dukungan.

Untuk mengonfigurasi AMI Anda sendiri untuk mendukung hibernasi, lihat [Konfigurasi 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 contoh

Anda harus menggunakan keluarga instance yang mendukung hibernasi.

- Tujuan umum: M3, M4, M5, M5a, M5ad, M5d, M6i, M6iD, M7i, M7i-flex, T2, T3, T3a
- Komputasi dioptimalkan: C3, C4, C5, C5d, C6i, C6id, C7a, C7i, C7i-flex
- Memori dioptimalkan: R3, R4, R5, R5a, R5ad, R5d, R7a, R7i, R7iZ
- Penyimpanan dioptimalkan: I3, i3en

Instans Nitro — Instans logam telanjang tidak didukung.

Untuk melihat tipe instans yang tersedia yang mendukung hibernasi di Wilayah tertentu

Tipe instans yang tersedia berbeda-beda menurut Wilayah. Untuk melihat tipe instans yang tersedia yang mendukung hibernasi di suatu Wilayah, gunakan perintah [describe-instance-types](#) dengan

parameter `--region`. Sertakan `--filters` parameter untuk cakupan hasil ke tipe instans yang mendukung hibernasi dan `--query` parameter untuk cakupan output ke nilai. `InstanceType`

```
aws ec2 describe-instance-types --filters Name=hibernation-supported,Values=true --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

Contoh Output

```
c3.2xlarge
c3.4xlarge
c3.8xlarge
c3.large
c3.xlarge
c4.2xlarge
c4.4xlarge
c4.8xlarge
...
```

Ukuran RAM instans

Instans Linux — Harus kurang dari 150 GB.

Instans Windows — Bisa sampai 16 GB. Untuk hibernasi instance Windows T3 atau T3a, kami merekomendasikan setidaknya 1 GB RAM.

Tipe volume root

Volume root harus berupa volume EBS, bukan volume penyimpanan instans.

Ukuran volume akar

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 dienkrpsi untuk memastikan perlindungan konten sensitif yang ada di memori pada saat hibernasi. Ketika data RAM dipindahkan ke volume root EBS, data itu selalu dienkrpsi. Enkripsi volume root diberlakukan saat peluncuran instans.

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 akun AWS 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 EC2 yang didukung EBS terenkripsi dari AMI yang tidak terenkripsi dan juga mengaktifkan hibernasi pada saat yang bersamaan. Untuk informasi selengkapnya, lihat [Menggunakan enkripsi dengan AMI yang didukung EBS](#).
- AMI terenkripsi – 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 lebih lanjut, lihat [Mengkripsikan 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.

Konfigurasi AMI Linux untuk mendukung hibernasi

AMI Linux berikut mendukung hibernasi, tetapi untuk hibernasi instance yang diluncurkan dengan salah satu AMI ini, konfigurasi tambahan diperlukan sebelum Anda dapat hibernasi instance.

Konfigurasi tambahan diperlukan untuk:

- [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 informasi selengkapnya, lihat [Memperbarui perangkat lunak instans di instans Amazon Linux 2 Anda](#).

Tidak diperlukan konfigurasi tambahan untuk AMI berikut karena sudah dikonfigurasi untuk mendukung hibernasi:

- AMI AL2023 yang dirilis pada 20/09/2023 atau setelahnya
- AMI lengkap Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya
- AMI Amazon Linux 2018.03 yang dirilis 16.11.2018 atau setelahnya
- AMI Ubuntu 22.04.2 LTS (Jammy Jellyfish) dirilis dengan nomor seri 20230303 atau setelahnya
- AMI Ubuntu 20.04 LTS (Focal Fossa) dirilis dengan nomor seri 20210820 atau setelahnya
- AMI Ubuntu 18.04 LTS (Bionic Beaver) dirilis dengan nomor seri 20190722.1 atau setelahnya

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 AMI yang didukung Amazon EBS-Backed](#).

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 AMI yang didukung Amazon EBS-Backed](#).

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 yang lebih baru, dan `grub2` ke `2.04-1ubuntu26.13` atau lebih baru.

```
[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 mengkonfigurasi Ubuntu 16.04 LTS untuk mendukung hibernasi, Anda perlu menginstal paket linux-aws-hwe kernel versi 4.15.0-1058-aws atau yang lebih baru 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.

Topik

- [Aktifkan hibernasi pada Instans Sesuai Permintaan](#)
- [Aktifkan hibernasi untuk Instans Spot](#)
- [Untuk melihat apakah instans diaktifkan untuk hibernasi](#)

Aktifkan hibernasi pada Instans Sesuai Permintaan

Gunakan salah satu metode berikut guna mengaktifkan hibernasi untuk Instans Sesuai Permintaan Anda.

New console

Untuk mengaktifkan hibernasi pada Instans Sesuai Permintaan

1. Ikuti prosedur untuk [meluncurkan instans](#), tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk mengaktifkan hibernasi.
2. Untuk mengaktifkan hibernasi, konfigurasi 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 contoh](#).
 - c. Pada Konfigurasi 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 AWS Wilayah ini, Ya dipilih.
 - Untuk Kunci KMS, pilih kunci enkripsi untuk volume. Jika Anda mengaktifkan enkripsi secara default di AWS Wilayah ini, kunci enkripsi default dipilih.

Untuk informasi selengkapnya tentang prasyarat volume root, lihat [Prasyarat untuk hibernasi instans Amazon 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Old console

Untuk mengaktifkan hibernasi pada Instans Sesuai Permintaan

1. Ikuti prosedur [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).
2. Di halaman Pilih Amazon Machine Image (AMI), pilih AMI yang mendukung hibernasi. Untuk informasi lebih lanjut tentang AMI yang didukung, lihat [Prasyarat untuk hibernasi instans Amazon EC2](#).
3. Di halaman Pilih Tipe Instans, pilih satu tipe instans yang didukung, lalu pilih Berikutnya: Konfigurasi Detail Instans. Untuk informasi tentang tipe instans yang didukung, lihat [Prasyarat untuk hibernasi instans Amazon EC2](#).
4. Di halaman Konfigurasi Detail Instans, untuk Perilaku Berhenti - Hibernasi, pilih kotak centang Aktifkan hibernasi sebagai perilaku berhenti tambahan.
5. Di halaman Tambahkan Penyimpanan, untuk volume root, tentukan informasi berikut:
 - 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 Enkripsi, pilih kunci enkripsi untuk volume. Jika Anda mengaktifkan enkripsi secara default di AWS Wilayah ini, kunci enkripsi default dipilih.

Untuk informasi selengkapnya tentang prasyarat untuk volume root, lihat [Prasyarat untuk hibernasi instans Amazon EC2](#) .

6. Lanjutkan seperti yang diminta oleh wizard. Setelah Anda selesai meninjau opsi di halaman Peluncuran Instans Peninjauan, pilih Luncurkan. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).

AWS CLI

Untuk mengaktifkan hibernasi pada Instans Sesuai Permintaan

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  
    }  
  }  
]
```

Note

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 AWS Wilayah ini, Anda dapat menghilangkannya `"Encrypted": true`.

PowerShell

Untuk mengaktifkan hibernasi untuk Instans Sesuai Permintaan menggunakan AWS Tools for Windows PowerShell

Gunakan [New-EC2Instance](#) perintah untuk meluncurkan sebuah 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`.

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType m5.large `
    -BlockDeviceMappings $ebs_encrypt `
    -HibernationOptions_Configured $true `
    -MinCount 1 `
    -MaxCount 1 `
    -KeyName MyKeyPair
```

Note

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

Gunakan salah satu metode berikut guna mengaktifkan hibernasi untuk Instans Spot Anda. Untuk informasi selengkapnya tentang hibernasi instans Spot saat interupsi, lihat [Interupsi Instans Spot](#).

Console

Anda dapat menggunakan wizard peluncuran instans di konsol Amazon EC2 guna mengaktifkan hibernasi untuk Instans Spot.

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, konfigurasi 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 contoh](#).
 - c. Pada Konfigurasi 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 AWS Wilayah ini, Ya dipilih.
 - Untuk Kunci KMS, pilih kunci enkripsi untuk volume. Jika Anda mengaktifkan enkripsi secara default di AWS Wilayah ini, kunci enkripsi default dipilih.

Untuk informasi selengkapnya tentang prasyarat volume root, lihat [Prasyarat untuk hibernasi instans Amazon 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

AWS CLI

Anda dapat mengaktifkan hibernasi untuk instans Spot menggunakan perintah AWS CLI [run-instances](#).

Untuk mengaktifkan hibernasi untuk Instans Spot menggunakan parameter **hibernation-options**

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
  }
}
```

Note

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 AWS Wilayah ini, Anda dapat menghilangkannya "Encrypted": true.

PowerShell

Untuk mengaktifkan hibernasi untuk Instance Spot menggunakan AWS Tools for Windows PowerShell

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`.

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> 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}
)

```

Note

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.

Ada beberapa cara di mana Anda dapat mengaktifkan hibernasi untuk Instans Spot. Untuk informasi selengkapnya, lihat [Menentukan perilaku interupsi](#).

Untuk melihat apakah instans diaktifkan untuk hibernasi

Gunakan instruksi berikut untuk melihat apakah sebuah instans 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 instans dan, pada tab Detail, di bagian Detail instans, periksa Perilaku berhenti - hibernasi. 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 menggunakan AWS Tools for Windows PowerShell

Gunakan [Get-EC2Instance](#) perintah dan tentukan `-Filter @{"Name="hibernation-options.configured"; Value="true"}` parameter untuk memfilter instance yang diaktifkan untuk hibernasi.

```
(Get-EC2Instance -Filter @{"Name="hibernation-options.configured";  
  Value="true"}).Instances
```

Outputnya mencantumkan instans EC2 yang diaktifkan untuk hibernasi.

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 [the section called "Hubungkan dengan SSH dari Linux atau macOS"](#).
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.

```
[ec2-user ~]$ sudo update-grub
```

5. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

6. Jalankan perintah berikut untuk mengonfirmasi bahwa `nokaslr` telah ditambahkan.

```
[ec2-user ~]$ cat /proc/cmdline
```

Output dari perintah harus menyertakan opsi `nokaslr`.

Hibernasi instans Amazon EC2

Anda dapat memulai hibernasi pada instans Sesuai Permintaan atau instans Spot jika instans tersebut merupakan instans yang didukung EBS, [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 Amazon 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 yang didukung EBS

Gunakan perintah [stop-instances](#) dan tentukan parameter `--hibernate`.

```
aws ec2 stop-instances \  
  --instance-ids i-1234567890abcdef0 \  
  --hibernate
```

PowerShell

Untuk hibernasi sebuah instance menggunakan AWS Tools for Windows PowerShell

Gunakan [Stop-EC2Instance](#) perintah dan tentukan `-Hibernate $true` parameternya.

```
Stop-EC2Instance \  
  -InstanceId i-1234567890abcdef0 \  
  -Hibernate $true
```

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.

Klien. `UserInitiatedHibernasi`: Hibernasi yang dimulai pengguna menunjukkan bahwa Anda memulai hibernasi pada Instance On-Demand atau 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 di keluaran menunjukkan bahwa hibernasi telah dimulai pada Instans Sesuai Permintaan atau Instans Spot.

```
"StateReason": {  
  "Code": "Client.UserInitiatedHibernate"  
}
```

PowerShell

Untuk melihat apakah hibernasi diinisiasi pada sebuah instans menggunakan AWS Tools for Windows PowerShell

Gunakan [Get-EC2Instance](#) perintah dan tentukan `state-reason-code` filter untuk melihat contoh di mana hibernasi dimulai.

```
Get-EC2Instance \  
  -Filter @{Name="state-reason-code";Value="Client.UserInitiatedHibernate"}
```

Keluaran mencantumkan instans EC2 tempat hibernasi dimulai.

Memulai instans Amazon EC2 yang hibernasi

Mulai instans hibernasi dengan memulainya dengan cara yang sama seperti Anda memulai instans yang dihentikan.

Note

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 instance hibernasi menggunakan AWS Tools for Windows PowerShell

Gunakan perintah [Start-EC2Instance](#).

```
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](#)
- [Terlalu lama untuk transisi dari `stopping` ke `stopped`, dan status memori tidak dipulihkan setelah dimulai](#)
- [Instans "macet" dalam status berhenti](#)
- [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.

Terlalu lama untuk transisi dari **stopping** ke **stopped**, dan status memori tidak dipulihkan setelah dimulai

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.

Contoh 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 Amazon EC2](#). Jika Anda menggunakan AMI Linux Anda sendiri, pastikan Anda mengikuti instruksi untuk [Konfigurasi AMI Linux untuk mendukung hibernasi](#).

Windows Server 2016 dan seterusnya

Periksa log peluncuran EC2 dan cari pesan yang terkait dengan hibernasi. Untuk mengakses log peluncuran EC2, [hubungankan](#) 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, [hubungankan](#) 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 instans Windows

Jika Anda menggunakan instans Windows T3 atau T3a dengan RAM kurang dari 1 GB, coba tingkatkan ukuran instans menjadi yang memiliki setidaknya 1 GB RAM.

Instans "macet" dalam status berhenti

Jika Anda menghibernasi instans Anda dan instans tersebut tampak "macet" di status `stopping`, Anda dapat menghentikannya secara paksa. Untuk informasi selengkapnya, lihat [Pemecahan masalah penghentian instans Anda](#).

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 instance 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`

Menyalakan ulang instans Anda

Sebuah instans yang melakukan boot ulang setara dengan boot 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

Melakukan boot ulang pada sebuah instans tidak akan memulai periode tagihan instans baru (dengan biaya minimum satu menit), tidak seperti [menghentikan dan memulai](#) instans Anda.

Kami mungkin menjadwalkan instans Anda untuk boot ulang untuk pemeliharaan yang diperlukan, seperti untuk menerapkan pembaruan yang memerlukan boot ulang. Anda tidak perlu melakukan tindakan apa pun; kami menyarankan Anda menunggu booting ulang terjadi dalam jendela yang dijadwalkan. Untuk informasi selengkapnya, lihat [Peristiwa terjadwal untuk instans Anda](#).

Kami menyarankan Anda untuk menggunakan konsol Amazon EC2, alat baris perintah, atau API Amazon EC2 untuk booting ulang instans Anda alih-alih menjalankan perintah boot ulang sistem operasi dari instans Anda. Jika Anda menggunakan konsol Amazon EC2, alat baris perintah, atau API Amazon EC2 untuk mem-boot ulang instans Anda, kami melakukan boot ulang paksa jika instans tidak mati dengan bersih dalam beberapa menit. Jika Anda menggunakan, AWS CloudTrail kemudian menggunakan Amazon EC2 untuk melakukan boot ulang instans Anda dan juga membuat catatan API saat instans Anda di-boot ulang.

Instans Windows

Jika Windows menginstal pembaruan pada instans Anda, kami menyarankan agar Anda tidak melakukan boot ulang atau mematikan instans Anda menggunakan konsol Amazon EC2 atau baris perintah hingga 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 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 instans state, Reboot instans.

Atau, pilih instans dan pilih Tindakan, Kelola status instans. Di layar yang terbuka, pilih Reboot, lalu Ubah status.

4. Pilih Boot ulang ketika diminta untuk konfirmasi.

Instans tetap dalam status `running`.

Command line

Untuk melakukan boot ulang instans

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [reboot-instances](#) (AWS CLI)
- [Restart-EC2Instance](#) (AWS Tools for Windows PowerShell)

Untuk menjalankan eksperimen injeksi kesalahan terkontrol

Anda dapat menggunakan AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons ketika instance Anda di-boot ulang. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS Fault Injection Service](#).

Mengakhiri instans Amazon EC2

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 tambahan menggunakan AMI yang sama. Jika Anda lebih suka menghentikan atau hibernasi sebuah instance, lihat [Hentikan dan mulai instans Amazon EC2](#) atau [Hibernasi instans Amazon EC2 Anda](#) Untuk informasi selengkapnya, lihat [Perbedaan antara boot ulang, penghentian, hibernasi, dan pengakhiran](#).

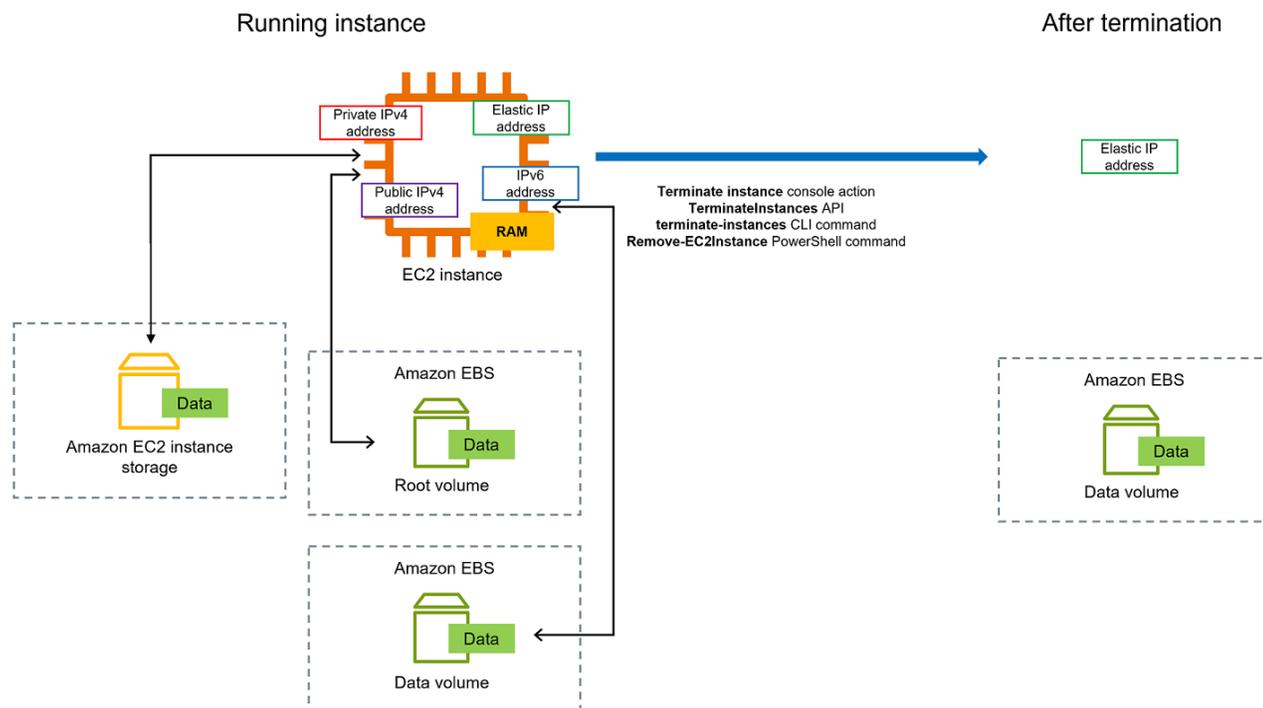
Daftar Isi

- [Cara kerja penghentian instance](#)
- [Akhiri instans](#)
- [Memecahkan masalah pengakhiran instans](#)
- [Aktifkan perlindungan pengakhiran](#)
- [Mengubah perilaku pematian yang diinisiasi oleh instans](#)
- [Pertahankan data saat instans diakhiri](#)

Cara kerja penghentian instance

Ketika Anda menghentikan sebuah instance, perubahan terdaftar pada tingkat OS instance, beberapa sumber daya hilang, dan beberapa sumber daya tetap ada.

Diagram berikut menunjukkan apa yang hilang dan apa yang bertahan ketika instans Amazon EC2 dihentikan. Ketika sebuah instance berakhir, data pada volume penyimpanan instans apa pun dan data yang disimpan RAM instance dihapus. Alamat IP Elastis apa pun yang terkait dengan instance terlepas. Untuk volume Amazon EBS dan data pada volume tersebut, hasilnya bergantung pada pengaturan Hapus pada penghentian untuk volume. Secara default, volume root dihapus dan volume data dipertahankan.



Pertimbangan

- Ketika sebuah instans berakhir, data pada setiap volume penyimpanan instans yang terkait dengan instans tersebut akan dihapus.
- Secara default, volume perangkat root Amazon EBS secara otomatis dihapus saat instans diakhiri. Namun, volume EBS tambahan apa pun yang Anda lampirkan saat peluncuran, atau volume EBS apa pun yang Anda lampirkan ke instans yang sudah ada akan tetap ada bahkan setelah instans berakhir. Untuk informasi selengkapnya, lihat [Pertahankan data saat instans diakhiri](#).

Note

Setiap volume yang tidak dihapus setelah pengakhiran instans akan terus dikenai biaya.

- Untuk mencegah instance dihentikan secara tidak sengaja oleh seseorang, [aktifkan perlindungan penghentian](#).
- Untuk mengontrol apakah instance berhenti atau berakhir saat shutdown dimulai dari instance, ubah perilaku shutdown yang [dimulai instance](#).
- Jika Anda menjalankan skrip pada pengakhiran instans, instans Anda mungkin mengalami pengakhiran yang tidak normal karena kami tidak memiliki cara untuk memastikan bahwa skrip

penonaktifan berjalan. Amazon EC2 mencoba menutup instans dengan bersih dan menjalankan skrip shutdown sistem apa pun; namun, kejadian tertentu (seperti kegagalan perangkat keras) dapat mencegah skrip pematian sistem ini berjalan.

Apa yang terjadi ketika Anda menghentikan sebuah instance

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.
- Instance akan mati 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.
- Data yang disimpan di volume perangkat root Amazon EBS jika atribut `DeleteOnTermination` diatur ke `true`.

Sumber daya yang bertahan

- Data yang disimpan di volume Amazon EBS tambahan yang dilampirkan saat peluncuran atau setelah peluncuran instans.

Uji respons aplikasi terhadap pengakhiran instans

Anda dapat menggunakan AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons ketika instance Anda dihentikan. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS Fault Injection Service](#).

Akhiri instans

Anda dapat menghentikan instance kapan saja.

Console

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, lalu pilih Status instans, Akhiri instans.
4. Pilih Akhiri saat diminta untuk konfirmasi.
5. Setelah Anda menghentikan sebuah instance, instans tetap terlihat untuk sementara waktu, dengan status. `terminated`

Jika penghentian gagal atau jika instance yang dihentikan terlihat selama lebih dari beberapa jam, lihat [Instans yang dihentikan masih ditampilkan](#).

Command line

Untuk mengakhiri instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [terminate-instances](#) (AWS CLI)
- [Remove-EC2Instance](#) (AWS Tools for Windows PowerShell)

Memecahkan masalah pengakhiran instans

Pemohon harus memiliki izin untuk menelepon `ec2:TerminateInstances`. 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](#).

Anda tidak dapat menghentikan instance jika perlindungan terminasi diaktifkan. Untuk informasi selengkapnya, lihat [perlindungan penghentian](#).

Jika instans Anda ada dalam status `shutting-down` lebih lama dari biasanya, instans harus dibersihkan (diakhiri) oleh proses otomatis dalam layanan Amazon EC2. Untuk informasi selengkapnya, lihat [Penghentian instans yang tertunda](#).

Aktifkan perlindungan pengakhiran

Untuk mencegah instans dari pengakhiran secara tidak sengaja, Anda dapat mengaktifkan perlindungan pengakhiran untuk instans. `DisableApiTermination` atribut mengontrol apakah instance dapat dihentikan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau API. Secara default, perlindungan terminasi dinonaktifkan untuk instans Anda, yang berarti instans Anda dapat dihentikan menggunakan AWS Management Console, AWS CLI, atau API. Anda dapat mengatur nilai atribut ini saat meluncurkan instans, ketika instans berjalan, atau ketika instans dihentikan (untuk instans yang didukung oleh Amazon EBS).

Atribut `DisableApiTermination` tidak mencegah Anda dari pengakhiran instans dengan memulai pematian dari instans tersebut (menggunakan perintah sistem operasi untuk pematian sistem) saat atribut `InstanceInitiatedShutdownBehavior` diatur. Untuk informasi selengkapnya, lihat [Mengubah perilaku pematian yang diinisiasi oleh instans](#).

Pertimbangan

- Mengaktifkan perlindungan terminasi tidak AWS mencegah penghentian instance ketika ada [acara terjadwal](#) untuk menghentikan instance.
- 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.

Untuk mengaktifkan perlindungan pengakhiran sebuah instans pada waktu peluncuran

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di dasbor, pilih Luncurkan instans dan ikuti petunjuk di wizard.
3. Di halaman Konfigurasi Detail Instans, pilih kotak centang Aktifkan perlindungan pengakhiran.

Untuk mengaktifkan perlindungan pengakhiran untuk instans yang berjalan atau berhenti

1. Pilih instans, dan pilih Tindakan, Pengaturan Instans, Ubah Perlindungan Pengakhiran.

2. Pilih Ya, Aktifkan.

Untuk menonaktifkan perlindungan pengakhiran untuk instans yang berjalan atau berhenti

1. Pilih instans, dan pilih Tindakan, Pengaturan Instans, Ubah Perlindungan Pengakhiran.
2. Pilih Ya, Nonaktifkan.

Untuk mengaktifkan atau menonaktifkan perlindungan pengakhiran menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

Mengakhiri beberapa instans dengan perlindungan terminasi

Jika Anda menghentikan beberapa instans di beberapa Availability Zone dalam permintaan yang sama, dan satu atau beberapa instance yang ditentukan diaktifkan untuk perlindungan penghentian, permintaan akan 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 contoh berikut di dua Availability Zone.

Instans	Zona Ketersediaan	Perlindungan pengakhiran
Contoh 1	AZ	Disabled
Contoh 2		Disabled
Contoh 3	AZ B	Enabled

Instans	Zona Ketersediaan	Perlindungan pengakhiran
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:

- Instance 1 dan Instance 2 berhasil dihentikan karena tidak ada instance yang diaktifkan untuk perlindungan terminasi.
- Instance 3 dan Instance 4 gagal dihentikan karena Instance 3 diaktifkan untuk perlindungan terminasi.

Mengubah perilaku pematian yang diinisiasi oleh instans

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.

Anda dapat mengubah atribut `InstanceInitiatedShutdownBehavior` menggunakan konsol Amazon EC2 atau baris perintah.

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.

Command line

Untuk mengubah perilaku pematian yang dinisiasi instans

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

Pertahankan data saat instans diakhiri

Bergantung pada kasus penggunaan, Anda mungkin ingin menyimpan data di volume penyimpanan instans atau volume Amazon EBS saat instans Amazon EC2 diakhiri. Data pada volume penyimpanan instans hilang saat instans diakhiri. Jika Anda harus 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. Untuk informasi selengkapnya, lihat [Opsinya penyimpanan untuk instans Amazon EC2 Anda](#).

Untuk data di volume Amazon EBS, Amazon EC2 menggunakan nilai atribut `DeleteOnTermination` untuk setiap volume Amazon EBS yang dilampirkan guna menentukan apakah akan mempertahankan atau menghapus volume tersebut.

Nilai default untuk atribut `DeleteOnTermination` berbeda-beda bergantung pada apakah volume tersebut adalah volume root dari instans atau volume non-root yang terpasang ke instans.

Volume root

Secara default, saat Anda meluncurkan instance, `DeleteOnTermination` atribut untuk volume root dari sebuah instance disetel ke `true`. Oleh karena itu, default-nya adalah menghapus volume root dari instans saat instans tersebut berakhir.

Volume non-root

Secara default, saat Anda melampirkan volume EBS non-root ke sebuah instance, `DeleteOnTermination` atributnya disetel ke `false`. Oleh karena itu, default-nya adalah untuk mempertahankan volume ini.

Note

Setelah instans berakhir, Anda dapat mengambil snapshot dari volume yang dipertahankan atau melampirkannya ke instans lain. Anda harus menghapus volume agar tidak dikenai biaya lebih lanjut.

Atribut `DeleteOnTermination` dapat diatur oleh pembuat AMI serta oleh orang yang meluncurkan instans. Saat atribut diubah oleh pembuat AMI atau oleh orang yang meluncurkan instans, pengaturan baru menggantikan pengaturan default AMI asli. Kami menyarankan Anda untuk memverifikasi pengaturan default untuk atribut `DeleteOnTermination` setelah Anda meluncurkan sebuah instans dengan AMI.

Untuk memverifikasi apakah volume Amazon EBS akan dihapus saat pengakhiran instans, lihat detail untuk volume di panel detail instans. Pada tab Penyimpanan, pada Perangkat blok, gulir ke kanan untuk melihat pengaturan Hapus saat pengakhiran untuk volume.

- Jika Ya, volume akan dihapus ketika instans diakhiri.
- Jika Tidak, volume tidak akan dihapus ketika instans diakhiri. Setiap volume yang tidak dihapus setelah pengakhiran instans akan terus dikenai biaya.

Ubah volume root untuk bertahan saat peluncuran

Dengan konsol, Anda dapat mengubah atribut `DeleteOnTermination` saat Anda meluncurkan suatu contoh. Untuk mengubah atribut ini untuk instans yang sedang berjalan, Anda harus menggunakan baris perintah.

Gunakan salah satu metode berikut untuk mengubah volume root agar tetap ada saat peluncuran.

Console

Mengubah volume root agar tetap ada saat peluncuran menggunakan konsol

1. Ikuti prosedur untuk [meluncurkan instans](#), tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut guna mengubah volume root agar tetap ada.
2. Di bawah Penyimpanan (volume), perluas informasi di bawah volume root.
3. Untuk Hapus saat pengakhiran, pilih Tidak
4. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Command line

Untuk mengubah volume root instans agar tetap ada saat peluncuran menggunakan baris perintah

Saat Anda meluncurkan instans yang didukung EBS, Anda dapat menggunakan salah satu dari perintah berikut untuk mengubah volume perangkat root menjadi persisten. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [run-instances](#) (AWS CLI)
- [New-EC2Instance](#) (AWS Tools for Windows PowerShell)

Dalam pemetaan perangkat blok untuk volume yang ingin Anda pertahankan, sertakan `--DeleteOnTermination`, dan tentukan `false`.

Misalnya, untuk mempertahankan volume, tambahkan opsi berikut ke perintah `run-instances` Anda:

```
--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
}
]
```

Ubah volume root dari instance yang sedang berjalan untuk bertahan

Anda dapat menggunakan salah satu dari perintah berikut untuk mengubah volume perangkat root dari instans yang didukung EBS yang berjalan agar persisten. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

Sebagai contoh, gunakan perintah berikut:

```
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
    }
  }
]
```

Pensiun instans

Sebuah instance dijadwalkan untuk dihentikan ketika AWS mendeteksi kegagalan yang tidak dapat diperbaiki dari perangkat keras yang mendasari yang menjadi tuan rumah instance. Perangkat root instance menentukan perilaku pensiun instance:

- Jika perangkat root instans Anda adalah volume Amazon EBS, instans akan dihentikan, dan Anda dapat memulainya lagi kapan saja. Memulai instans yang dihentikan, migrasikan ke perangkat keras baru.
- Jika perangkat root instance Anda adalah volume penyimpanan instance, instance akan dihentikan, dan tidak dapat digunakan lagi.

Untuk informasi selengkapnya tentang tipe peristiwa instans, lihat [Peristiwa terjadwal untuk instans Anda](#).

Daftar Isi

- [Identifikasi instans yang dijadwalkan untuk pensiun](#)
- [Tindakan yang harus diambil untuk instans yang didukung EBS 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 instans yang dijadwalkan untuk pensiun menggunakan konsol Amazon EC2 atau baris perintah.

Important

Jika sebuah instans dijadwalkan untuk pensiun, kami menyarankan Anda untuk mengambil tindakan sesegera mungkin karena instans tersebut mungkin tidak dapat dijangkau. (Notifikasi email yang Anda terima menyatakan sebagai berikut: "Karena degradasi ini, instans Anda mungkin sudah tidak dapat dijangkau.") Untuk informasi selengkapnya tentang rekomendasi tindakan yang harus Anda lakukan, lihat [Check if your instance is reachable](#).

Cara untuk mengidentifikasi instans yang dijadwalkan untuk pensiun

- [Notifikasi email](#)
- [Identifikasi konsol](#)

Notifikasi email

Jika instans Anda dijadwalkan untuk pensiun, Anda akan menerima email sebelum peristiwa itu disertai dengan ID instans dan tanggal pensiun.

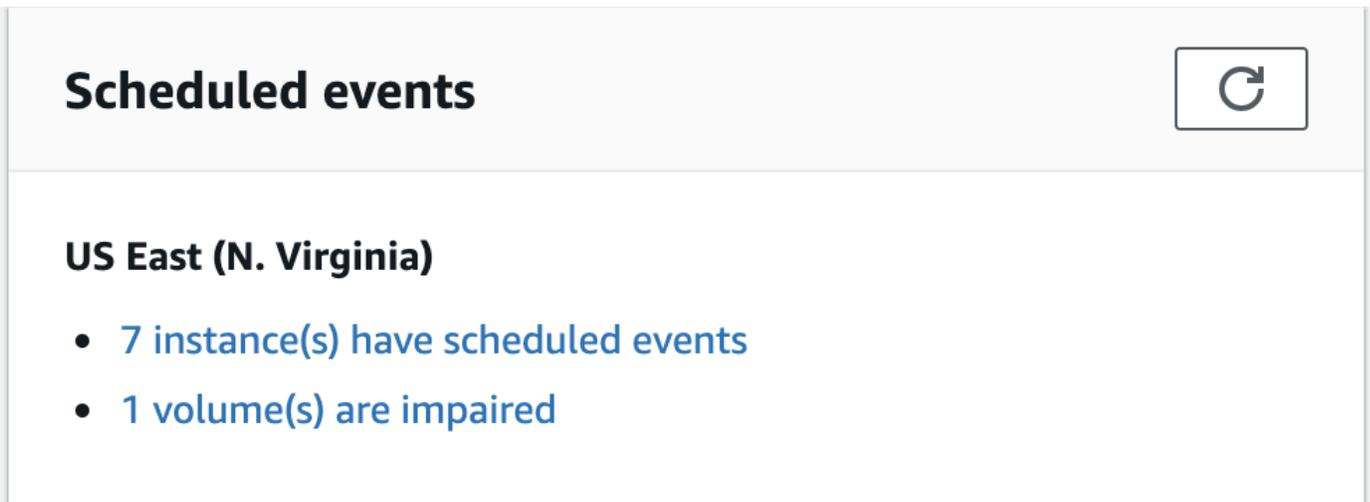
Email dikirim ke pemegang akun utama dan kontak operasi. Untuk informasi selengkapnya, lihat [Menambahkan, mengubah, atau menghapus kontak alternatif](#) di Panduan Pengguna AWS Billing .

Identifikasi konsol

Jika Anda menggunakan akun email yang tidak Anda periksa secara teratur untuk notifikasi pensiun instans, Anda dapat menggunakan konsol Amazon EC2 atau baris perintah untuk menentukan apakah ada instans Anda yang dijadwalkan untuk pensiun.

Untuk mengidentifikasi instans yang dijadwalkan untuk pensiun menggunakan konsol

1. Buka konsol Amazon 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.

Untuk mengidentifikasi instans yang dijadwalkan untuk pensiun menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-instance-status](#) (AWS CLI)
- [Get-EC2InstanceStatus](#) (AWS Tools for Windows PowerShell)

Tindakan yang harus diambil untuk instans yang didukung EBS 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 instance Linux, jika Anda tidak yakin apakah instans Anda didukung oleh EBS atau instans store, lihat. [Tentukan jenis perangkat root dari instance Linux 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, Anda harus merencanakan untuk menghentikan/memulai instans Anda 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 [Hentikan dan mulai 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 [penghentian/mulai](#) 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 AMI yang didukung EBS dari instans 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 AMI yang didukung Amazon EBS-Backed](#).

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 Tindakan, Luncurkan. Ikuti wizard untuk meluncurkan instans Anda. Untuk informasi selengkapnya tentang setiap langkah dalam wizard, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

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 instans yang didukung penyimpanan instans Anda melewati tanggal pensiun, instans tersebut diakhiri dan Anda tidak dapat memulihkan instans atau data apa pun yang disimpan di dalamnya. Terlepas dari perangkat root instans Anda, data pada volume penyimpanan instans akan hilang saat instans pensiun, meskipun volume tersebut dipasang ke instans yang didukung 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, lihat [Memecahkan masalah instans yang tidak dapat dijangkau](#). AWS akan mengakhiri instance Anda pada tanggal pensiun yang

dijadwalkan, jadi, untuk contoh yang tidak terjangkau, Anda dapat segera [menghentikan](#) instance itu sendiri.

Luncurkan instans pengganti

Buat AMI yang didukung penyimpanan instans dari instans Anda menggunakan alat AMI, seperti yang dijelaskan di [Buat AMI Linux yang didukung penyimpanan instans](#). Dari konsol Amazon EC2, pilih AMI baru Anda lalu pilih Tindakan, Luncurkan. Ikuti wizard untuk meluncurkan instans Anda. Untuk informasi selengkapnya tentang setiap langkah dalam wizard, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Ubah instans Anda menjadi instans yang didukung EBS

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. Untuk informasi selengkapnya, lihat [Konversikan AMI yang didukung penyimpanan instans menjadi AMI yang didukung Amazon EBS](#).

Ketahanan contoh

Important

Informasi berikut berlaku untuk mengonfigurasi kemampuan terkait pemulihan pada instans yang sehat. Jika saat ini Anda mengalami kesulitan mengakses instans Anda, lihat [Memecahkan masalah](#) instans EC2.

Jika instans tidak tersedia karena masalah perangkat keras yang mendasarinya, ada dua mekanisme yang dapat Anda konfigurasi misalnya ketahanan yang dapat memulihkan ketersediaan — pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatchtindakan Amazon. AWS Proses ini disebut pemulihan instance.

Setidaknya satu mekanisme harus dikonfigurasi atau diaktifkan terlebih dahulu dengan sumber daya yang didukung agar proses pemulihan instans terjadi. Secara default, pemulihan otomatis yang disederhanakan diaktifkan untuk instance yang didukung saat diluncurkan.

Topik

- [Ikhtisar pemulihan instance](#)

- [Alternatif pemulihan instans](#)
- [Konfigurasi pemulihan berbasis CloudWatch tindakan](#)
- [Konfigurasi pemulihan otomatis yang disederhanakan](#)

Ikhtisar pemulihan instance

Berikut ini adalah contoh masalah perangkat keras yang mendasari yang mungkin memerlukan pemulihan instance:

- Kehilangan konektivitas jaringan
- Kehilangan daya sistem
- Masalah perangkat lunak pada host fisik
- Masalah perangkat keras pada hosting fisik yang memengaruhi jangkauan jaringan

Instance yang dipulihkan identik dengan instance asli, termasuk:

- ID Instans
- Alamat IP publik, pribadi, dan Elastis
- Metadata instans
- Grup penempatan
- Volume EBS terlampir
- Zona Ketersediaan

Pemulihan instans yang berhasil akan muncul ke instance sebagai reboot yang tidak direncanakan. Dengan kata lain, konten yang disimpan dalam memori volatil akan hilang, data penyimpanan instance akan dihapus, dan uptime sistem operasi akan dimulai dari nol.

Untuk membantu melindungi dari kehilangan data, kami sarankan Anda secara teratur membuat cadangan data berharga. Untuk informasi selengkapnya tentang praktik terbaik pencadangan dan pemulihan untuk instans Amazon EC2, lihat [Praktik terbaik untuk Amazon EC2](#).

Alternatif pemulihan instans

Alternatif berikut untuk pemulihan instance dapat dipertimbangkan ketika mereka memenuhi kasus penggunaan instance Anda.

Grup Auto Scaling

Anda dapat menggunakan grup Auto Scaling untuk memungkinkan Anda mengelompokkan kumpulan instance bersama-sama untuk tujuan penskalaan dan ketersediaan. Jika instance dalam grup Auto Scaling menjadi tidak tersedia, instance akan secara otomatis diganti (tidak dipulihkan) oleh grup Auto Scaling. Untuk informasi selengkapnya, lihat [Apa itu Auto Scaling Amazon EC2?](#) di Panduan Pengguna Auto Scaling Amazon EC2.

Amazon EBS Multi-Lampiran

Anda dapat mengonfigurasi Amazon EBS Multi-Attach untuk instans agar beberapa instans dapat dihubungkan ke volume EBS yang sama. Ketika dikombinasikan dengan perangkat lunak yang sesuai, ini memungkinkan pengelompokan ketersediaan tinggi diaktifkan. Untuk contoh konfigurasi dengan instance Linux, lihat [Penyimpanan berkelompok yang disederhanakan: GFS2 di Amazon EBS Multi-Lampirkan volume yang diaktifkan](#) di Blog Penyimpanan. AWS

Konfigurasi pemulihan berbasis CloudWatch tindakan

Important

- Informasi berikut berlaku untuk mengonfigurasi kemampuan terkait pemulihan pada instans yang sehat. Jika saat ini Anda mengalami kesulitan mengakses instans Anda, lihat [Memecahkan masalah](#) instans EC2.
- Agar beban kerja Anda berfungsi dengan baik setelah pemulihan instans berhasil, instans Anda harus boot dan menerima lalu lintas tanpa memerlukan intervensi manual.

Anda dapat mengonfigurasi pemulihan berbasis CloudWatch tindakan Amazon untuk menambahkan tindakan pemulihan ke CloudWatch alarm Amazon. CloudWatch pemulihan berbasis tindakan bekerja dengan StatusCheckFailed_System metrik. CloudWatch pemulihan berbasis tindakan memberikan perincian waktu respons to-the-minute pemulihan 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 [Pemeriksaan status untuk instans Anda](#).

Pemulihan berbasis CloudWatch tindakan Amazon tidak beroperasi selama peristiwa layanan di AWS Health Dashboard. Untuk informasi selengkapnya, lihat [the section called “Pemecahan masalah kegagalan CloudWatch pemulihan berbasis tindakan”](#).

Topik

- [Persyaratan dan batasan untuk pemulihan berbasis CloudWatch tindakan](#)
- [Konfigurasi pemulihan berbasis CloudWatch tindakan](#)
- [Pemecahan masalah kegagalan CloudWatch pemulihan berbasis tindakan](#)

Persyaratan dan batasan untuk pemulihan berbasis CloudWatch tindakan

CloudWatch pemulihan berbasis tindakan dapat mencoba memulihkan instance jika:

- Ada di `running` negara bagian. Untuk informasi selengkapnya, lihat [the section called “Siklus hidup instans”](#).
- Penggunaan default (Sesuai Permintaan) atau dedicated penyewaan instance. Untuk informasi selengkapnya, lihat [the section called “Opsi pembelian instans”](#).
- Adalah jenis instans yang Amazon EC2 memiliki kapasitas yang tersedia. Dalam beberapa situasi, seperti pemadaman yang signifikan, kapasitas tidak cukup akan tersedia dan beberapa upaya pemulihan mungkin gagal.
- Tidak menggunakan penyewaan dedicated instance. Untuk Host Khusus Amazon EC2, Anda dapat menggunakan [Pemulihan Otomatis Host Khusus](#) untuk memulihkan instans yang tidak sehat secara otomatis.
- Tidak menggunakan Adaptor Kain Elastis.
- Bukan anggota grup Auto Scaling.
- Saat ini tidak sedang menjalani acara pemeliharaan terjadwal.
- Menggunakan salah satu jenis contoh berikut:
 - Tujuan umum: A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
 - Komputasi dioptimalkan: C3 | C4 | C5 | C5a | C5n | C6a | C6g | C6gN | C6i | C6in | C7a | C7g | C7gN | C7i | C7i-flex
 - Memori dioptimalkan: R3 | R4 | R5 | R5a | R5b | R5n | R6a | R6g | R6i | R6in | R7a | R7g | R7i | R7iZ | u-3tb1 | u-6tb1 | u-9tb1 | u-12tb1 | u-24tb1 | u7i-12tb | u77b dalam-16tb | u7in-24tb | u7in-32tb | X1 | x1e | x2IEZN

- Komputasi terakselerasi: G3 | G3s | G5g | Inf1 | P2 | P3 | VT1
- Komputasi kinerja tinggi: hPC6a | hPC7a | hPC7g
- Contoh logam: Salah satu jenis di atas dengan ukuran instans logam.
- Memiliki volume penyimpanan instans dan menggunakan salah satu jenis instance berikut: M3 | C3 | R3 | X1 | x1e | x2idn | x2iedn

Warning

- Data pada volume penyimpanan instance akan hilang jika instance dihentikan. Untuk informasi selengkapnya tentang menghentikan instance, lihat [the section called “Instans berhenti dan mulai”](#).
- Jika terjadi kegagalan pemeriksaan status sistem, penyimpanan instance dan blok data yang dipetakan perangkat mungkin hilang. Untuk jenis contoh ini, Anda dapat mempertimbangkan untuk menggunakan [the section called “Aktifkan perlindungan pengakhiran”](#).

Kami menyarankan Anda secara teratur membuat cadangan data berharga. Untuk informasi tentang praktik terbaik pencadangan dan pemulihan untuk Amazon EC2, lihat [Praktik terbaik untuk Amazon EC2](#).

Anda juga dapat menggunakan AWS Management Console atau AWS CLI untuk melihat jenis instance yang mendukung pemulihan berbasis CloudWatch tindakan.

Console

Untuk melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan Amazon

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 Pemulihan Otomatis: benar. Atau, saat Anda memasukkan karakter dan nama filter muncul, Anda dapat memilihnya.

Tabel tipe Instance menampilkan semua jenis instance yang mendukung pemulihan berbasis CloudWatch tindakan Amazon.

AWS CLI

Untuk melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan Amazon

Gunakan perintah [describe-instance-types](#).

```
aws ec2 describe-instance-types --filters Name=auto-recovery-supported,Values=true
--query "InstanceTypes[*].[InstanceType]" --output text | sort
```

Konfigurasi pemulihan berbasis CloudWatch tindakan

CloudWatch pemulihan berbasis tindakan bekerja dengan `StatusCheckFailed_System` metrik. CloudWatch pemulihan berbasis tindakan dikonfigurasi melalui CloudWatch konsol. Untuk mengatur pemulihan berbasis CloudWatch tindakan, lihat [Menambahkan tindakan pemulihan ke CloudWatch alarm](#) di Panduan CloudWatch Pengguna Amazon.

Pemecahan masalah kegagalan CloudWatch pemulihan berbasis tindakan

Masalah berikut dapat menyebabkan pemulihan instans Anda dengan pemulihan berbasis CloudWatch tindakan gagal:

- CloudWatch pemulihan berbasis tindakan tidak beroperasi selama acara layanan di AWS Health Dashboard. Anda mungkin tidak menerima notifikasi kegagalan pemulihan untuk peristiwa semacam itu. Untuk informasi ketersediaan layanan terbaru, lihat halaman Status [kesehatan layanan](#).
- Sementara, kapasitas perangkat keras pengganti tidak mencukupi.
- Instans telah mencapai tunjangan harian maksimum untuk upaya pemulihan. Instans Anda mungkin kemudian akan dipensiunkan jika pemulihan otomatis gagal dan degradasi perangkat keras ditentukan sebagai akar penyebab kegagalan pemeriksaan status sistem asli.

Jika kegagalan pemeriksaan status sistem instans tetap ada meskipun beberapa upaya pemulihan, lihat [Memecahkan masalah instance dengan pemeriksaan status gagal](#) untuk panduan tambahan.

Konfigurasi pemulihan otomatis yang disederhanakan

Important

- Informasi berikut berlaku untuk mengonfigurasi kemampuan terkait pemulihan pada instans yang sehat. Jika saat ini Anda mengalami kesulitan mengakses instans Anda, lihat [Memecahkan masalah](#) instans EC2.
- Agar beban kerja Anda berfungsi dengan baik setelah pemulihan instans berhasil, instans Anda harus boot dan menerima lalu lintas tanpa memerlukan intervensi manual.

Secara default, pemulihan otomatis yang disederhanakan memonitor semua instance berjalan yang didukung. Jika kegagalan pemeriksaan status sistem terdeteksi, pemulihan otomatis yang disederhanakan mencoba untuk memulihkan instance ke keadaan sehat. Pemulihan otomatis yang disederhanakan tidak beroperasi selama acara layanan di AWS Health Dashboard. Untuk informasi selengkapnya, lihat [the section called “Pemecahan masalah kegagalan pemulihan otomatis yang disederhanakan”](#).

Ketika peristiwa pemulihan otomatis yang disederhanakan terjadi, Anda akan menerima suatu AWS Health Dashboard acara. Untuk mengonfigurasi notifikasi untuk peristiwa ini, lihat [Memulai dengan Notifikasi Pengguna AWS](#) di Panduan Notifikasi Pengguna AWS Pengguna. Anda juga dapat menggunakan EventBridge aturan Amazon untuk memantau peristiwa pemulihan otomatis yang disederhanakan menggunakan kode peristiwa berikut:

- `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_SUCCESS` — peristiwa berhasil
- `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_FAILURE` — peristiwa yang gagal

Untuk informasi selengkapnya, lihat [EventBridge Aturan Amazon](#).

Topik

- [Persyaratan dan batasan untuk pemulihan otomatis yang disederhanakan](#)
- [Konfigurasi pemulihan otomatis yang disederhanakan](#)
- [Pemecahan masalah kegagalan pemulihan otomatis yang disederhanakan](#)

Persyaratan dan batasan untuk pemulihan otomatis yang disederhanakan

Pemulihan otomatis yang disederhanakan akan mencoba memulihkan instance jika:

- Ada di `running` negara bagian. Untuk informasi selengkapnya, lihat [the section called “Siklus hidup instans”](#).
- Penggunaan default (Sesuai Permintaan) atau dedicated penyewaan instance. Untuk informasi selengkapnya, lihat [the section called “Opsi pembelian instans”](#).
- Adalah jenis instans yang Amazon EC2 memiliki kapasitas yang tersedia. Dalam beberapa situasi, seperti pemadaman yang signifikan, kapasitas tidak cukup akan tersedia dan beberapa upaya pemulihan mungkin gagal.
- Tidak menggunakan penyewaan dedicated instance. Untuk Host Khusus Amazon EC2, Anda dapat menggunakan [Pemulihan Otomatis Host Khusus](#) untuk memulihkan instans yang tidak sehat secara otomatis.
- Tidak menggunakan Adaptor Kain Elastis.
- Bukan ukuran `metal` instance.
- Bukan anggota grup Auto Scaling.
- Saat ini tidak sedang menjalani acara pemeliharaan terjadwal.
- Tidak memiliki volume penyimpanan instance.
- Menggunakan salah satu jenis contoh berikut:
 - Tujuan umum: A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
 - Komputasi dioptimalkan: C3 | C4 | C5 | C5a | C5n | C6a | C6g | C6gN | C6i | C6in | C7a | C7g | C7gN | C7i | C7i-flex
 - Memori dioptimalkan: R3 | R4 | R5 | R5a | R5b | R5n | R6a | R6g | R6i | R6in | R7a | R7g | R7i | R7iZ | u-3tb1 | u-6tb1 | u-9tb1 | u-12tb1 | u-24tb1 | u7i-12tb | u77b dalam-16tb | u7in-24tb | u7in-32tb | X1 | x1e | x2IEZN
 - Komputasi terakselerasi: G3 | G3s | G5g | Inf1 | P2 | P3 | VT1
 - Komputasi kinerja tinggi: hPC6a | hPC7a | hPC7g

Warning

- Data pada volume penyimpanan instance akan hilang jika instance dihentikan. Untuk informasi selengkapnya tentang menghentikan instance, lihat [the section called “Instans berhenti dan mulai”](#).
- Jika terjadi kegagalan pemeriksaan status sistem, penyimpanan instance dan blok data yang dipetakan perangkat mungkin hilang. Untuk jenis contoh ini, Anda dapat mempertimbangkan untuk menggunakan [the section called “Aktifkan perlindungan pengakhiran”](#).

Kami menyarankan Anda secara teratur membuat cadangan data berharga. Untuk informasi tentang praktik terbaik pencadangan dan pemulihan untuk Amazon EC2, lihat [Praktik terbaik untuk Amazon EC2](#).

Konfigurasi pemulihan otomatis yang disederhanakan

Pemulihan otomatis yang disederhanakan diaktifkan secara default saat Anda meluncurkan instans yang didukung. Anda dapat mengatur perilaku pemulihan otomatis disabled selama atau setelah meluncurkan instance. defaultKonfigurasi tidak mengaktifkan pemulihan otomatis yang disederhanakan untuk jenis instans yang tidak didukung.

Console

Untuk menonaktifkan pemulihan otomatis simpel selama peluncuran instans

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 Dinonaktifkan.
4. Konfigurasi pengaturan peluncuran instans yang tersisa sesuai kebutuhan kemudian luncurkan instans.

Untuk menonaktifkan pemulihan otomatis yang disederhanakan untuk instans yang berjalan atau dihentikan

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 mengatur perilaku pemulihan otomatis ke **default** untuk instans yang berjalan atau dihentikan

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-instans](#).

```
aws ec2 run-instances \  
--image-id ami-1a2b3c4d \  
--instance-type t2.micro \  
--key-name MyKeyPair \  
--maintenance-options AutoRecovery=Disabled \  
[...]
```

Untuk menonaktifkan pemulihan otomatis yang disederhanakan untuk instans yang berjalan atau dihentikan

Gunakan perintah [modify-instance-maintenance-options](#).

```
aws ec2 modify-instance-maintenance-options \  
--instance-id i-0abcdef1234567890 \  
--auto-recovery disabled
```

Untuk mengatur perilaku pemulihan otomatis ke **default** untuk instans yang berjalan atau dihentikan

Gunakan perintah [modify-instance-maintenance-options](#).

```
aws ec2 modify-instance-maintenance-options \  
--instance-id i-0abcdef1234567890 \  
--auto-recovery default
```

Pemecahan masalah kegagalan pemulihan otomatis yang disederhanakan

Masalah berikut dapat menyebabkan pemulihan instans Anda dengan pemulihan otomatis yang disederhanakan gagal:

- Pemulihan otomatis yang disederhanakan tidak beroperasi selama acara layanan di AWS Health Dashboard. Anda mungkin tidak menerima notifikasi kegagalan pemulihan untuk peristiwa semacam itu. Untuk informasi ketersediaan layanan terbaru, lihat halaman Status [kesehatan layanan](#).
- Sementara, kapasitas perangkat keras pengganti tidak mencukupi.
- Instans telah mencapai tunjangan harian maksimum untuk upaya pemulihan. Instans Anda mungkin kemudian akan dipensiunkan jika pemulihan otomatis gagal dan degradasi perangkat keras ditentukan sebagai akar penyebab kegagalan pemeriksaan status sistem asli.

Jika kegagalan pemeriksaan status sistem instans tetap ada meskipun beberapa upaya pemulihan, lihat [Memecahkan masalah instance dengan pemeriksaan status gagal](#) untuk panduan tambahan.

Cara menggunakan metadata instans

Metadata instans adalah data tentang instans Anda yang dapat Anda gunakan untuk mengonfigurasi atau mengelola instans berjalan. Metadata instans dibagi menjadi beberapa [kategori](#), misalnya, nama host, peristiwa, dan grup keamanan.

Anda juga dapat menggunakan metadata instans untuk mengakses data pengguna yang Anda tentukan saat meluncurkan instans Anda. 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 instance menggunakan RunInstances panggilan yang sama, data pengguna tersedia untuk semua instance 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 [.Contoh Linux: Nilai indeks peluncuran AMI](#)

Instans EC2 juga dapat menyertakan data dinamis, seperti dokumen identitas instans yang dibuat saat instans diluncurkan. Untuk informasi selengkapnya, lihat [Kategori data dinamis](#).

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

- [Gunakan IMDSv2](#)
- [Mengonfigurasi opsi metadata instans](#)
- [Mengambil metadata instans](#)
- [Bekerja dengan data pengguna instans](#)
- [Mengambil data dinamis](#)
- [Kategori metadata instans](#)
- [Contoh Linux: Nilai indeks peluncuran AMI](#)
- [Dokumen identitas instans](#)
- [Peran identitas instans](#)

Gunakan IMDSv2

Anda dapat mengakses metadata instans dari instans yang sedang berjalan menggunakan salah satu metode berikut:

- Layanan Metadata Instans Versi 1 (IMDSv1) – metode permintaan/tanggapan
- Layanan Metadata Instans Versi 2 (IMDSv2) - metode berorientasi sesi

Secara default, Anda dapat menggunakan IMDSv1 atau IMDSv2, atau keduanya.

Anda dapat mengonfigurasi Layanan Metadata Instans (IMDS) pada setiap instans yang mengharuskan kode atau pengguna lokal untuk menggunakan IMDSv2. Saat Anda menentukan bahwa IMDSv2 harus digunakan, maka IMDSv1 tidak lagi berfungsi. Untuk informasi tentang cara mengonfigurasi instans Anda agar menggunakan IMDSv2, lihat [Mengonfigurasi opsi metadata instans](#).

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](#).

Untuk mengambil metadata instans, lihat [Mengambil metadata instans](#).

Topik

- [Bagaimana cara kerja Layanan Metadata Instans Versi 2](#)
- [Transisi ke penggunaan Layanan Metadata Instans Versi 2](#)
- [Menggunakan AWS SDK yang didukung](#)

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 [instans yang dibangun pada Sistem AWS Nitro](#) dan dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#).

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 (instance 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 permintaan 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](#).

Metode GET dan HEAD HTTP diizinkan dalam permintaan metadata instans IMDSv2. Permintaan PUT ditolak jika berisi header X-Forwarded-For.

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 [AWS CLI modify-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](#).

Transisi ke penggunaan Layanan Metadata Instans Versi 2

Saat bermigrasi ke IMDSv2, kami menyarankan Anda untuk menggunakan alat dan jalur transisi berikut.

Topik

- [Alat untuk membantu transisi ke IMDSv2](#)

- [Jalur yang disarankan untuk mengharuskan IMDSv2](#)

Alat untuk membantu transisi ke IMDSv2

Jika perangkat lunak Anda menggunakan IMDSv1, gunakan alat bantu berikut untuk membantu mengonfigurasi ulang perangkat lunak Anda untuk menggunakan IMDSv2.

AWS perangkat lunak

Versi terbaru dari AWS CLI dan AWS SDK mendukung IMDSv2. Untuk menggunakan IMDSv2, pastikan bahwa instans EC2 Anda memiliki CLI dan SDK versi terbaru. Untuk informasi tentang pembaruan CLI, lihat [Menginstal, memperbarui, dan menghapus instalasi AWS CLI](#) di Panduan Pengguna AWS Command Line Interface .

Semua paket perangkat lunak Amazon Linux 2 dan Amazon Linux 2023 mendukung IMDSv2. Di Amazon Linux 2023, IMDSv1 dinonaktifkan secara default.

Untuk versi AWS SDK minimum yang mendukung IMDSv2, lihat [Menggunakan AWS SDK yang didukung](#)

IMDS Package Analyzer

IMDS Packet Analyzer adalah alat sumber terbuka yang mengidentifikasi dan mencatat panggilan IMDSv1 dari fase boot instans Anda. Hal ini dapat membantu mengidentifikasi perangkat lunak yang membuat panggilan IMDSv1 pada instans EC2, sehingga Anda dapat menentukan dengan tepat apa yang perlu Anda perbarui agar instans Anda siap menggunakan IMDSv2 saja. Anda dapat menjalankan IMDS Packet Analyzer dari baris perintah atau menginstalnya sebagai layanan. Untuk informasi lebih lanjut, lihat [IMDS Packet Analyzer](#) di GitHub

CloudWatch

IMDSv2 menggunakan sesi yang didukung token, sementara IMDSv1 tidak. `MetadataNoToken` CloudWatch Metrik melacak jumlah panggilan ke Layanan Metadata Instans (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.

Setelah menonaktifkan IMDSv1, Anda dapat menggunakan `MetadataNoTokenRejected` CloudWatch 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 informasi selengkapnya, lihat [Metrik instans](#).

Pembaruan pada API dan CLI EC2

Untuk instance baru, Anda dapat menggunakan [RunInstances](#) API untuk meluncurkan instance baru yang memerlukan penggunaan IMDSv2. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans untuk instans baru](#).

Untuk instance yang ada, Anda dapat menggunakan [ModifyInstanceMetadataOptions](#) API untuk meminta penggunaan IMDSv2. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans untuk instans yang ada](#).

Untuk mengharuskan penggunaan IMDSv2 pada semua instans baru yang diluncurkan oleh grup Auto Scaling, grup Auto Scaling Anda dapat menggunakan templat 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. Untuk instans yang ada di grup Auto Scaling, Anda dapat menggunakan [ModifyInstanceMetadataOptions](#) API untuk meminta penggunaan IMDSv2 pada instans yang ada, atau menghentikan instance dan grup Auto Scaling akan meluncurkan instance pengganti baru dengan pengaturan opsi metadata instans yang ditentukan dalam templat peluncuran baru atau konfigurasi peluncuran.

Gunakan AMI yang mengonfigurasi IMDSv2 secara default

Saat meluncurkan instans, Anda dapat mengonfigurasinya secara otomatis untuk menggunakan IMDSv2 secara default (parameter `HttpTokens` diatur ke `required`) dengan meluncurkannya dengan AMI yang dikonfigurasi dengan parameter `ImdsSupport` yang diatur ke `v2.0`. Anda dapat mengatur parameter `ImdsSupport` ke `v2.0` saat mendaftarkan AMI menggunakan perintah CLI [register-image](#), atau Anda dapat memodifikasi AMI yang ada dengan menggunakan perintah CLI [modify-image-attribute](#). Untuk informasi selengkapnya, lihat [Konfigurasi AMI](#).

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 instance dikonfigurasi untuk menggunakan IMDSv2.
- Tidak dapat memodifikasi instance yang sedang berjalan 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 syarat tersebut, panggilan API atau CLI akan gagal dengan tanggapan `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 kredensial Peran EC2, Anda dapat menggunakan kunci kondisi baru 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 kredensial Peran EC2 yang diperoleh dari IMDSv1 akan menerima tanggapan `UnauthorizedOperation`. Hal yang sama dapat dicapai secara lebih luas dengan kondisi yang disyaratkan oleh SCP. Hal ini memastikan bahwa kredensial yang dikirim melalui IMDSv1 sebenarnya tidak bisa digunakan untuk memanggil API karena panggilan API yang tidak cocok dengan kondisi yang ditentukan akan menerima kesalahan `UnauthorizedOperation`.

Misalnya kebijakan IAM, lihat [.Cara menggunakan metadata instans](#) Untuk informasi lebih lanjut tentang SCP, lihat [Kebijakan Penolakan Layanan](#) di Panduan Pengguna AWS Organizations .

Jalur yang disarankan untuk mengharuskan IMDSv2

Menggunakan alat di atas, kami menyarankan Anda mengikuti jalur ini untuk transisi ke IMDSv2:

Langkah 1: Pada awal

Perbarui SDK, CLI, dan perangkat lunak Anda yang menggunakan kredensial Peran pada instans EC2 mereka untuk versi yang kompatibel dengan IMDSv2. Untuk informasi tentang pembaruan CLI, lihat [Memutakhirkan ke AWS CLI versi terbaru](#) di Panduan Pengguna AWS Command Line Interface .

Kemudian, ubah perangkat lunak Anda yang langsung mengakses metadata instans (dengan kata lain, yang tidak menggunakan SDK) menggunakan permintaan IMDSv2. Anda dapat menggunakan [IMDS Packet Analyzer](#) untuk mengidentifikasi perangkat lunak yang perlu Anda ubah untuk menggunakan permintaan IMDSv2.

Langkah 2: Lacak kemajuan transisi Anda

Lacak kemajuan transisi Anda dengan menggunakan CloudWatch metrik `MetadataNoToken`. Metrik ini melacak jumlah panggilan IMDSv1 ke IMDS di instans Anda. Untuk informasi selengkapnya, lihat [Metrik instans](#).

Langkah 3: Ketika tidak ada penggunaan IMDSv1

Saat CloudWatch metrik `MetadataNoToken` mencatat nol penggunaan IMDSv1, instance Anda siap untuk sepenuhnya dialihkan ke penggunaan IMDSv2. Pada tahap ini, Anda dapat melakukan hal berikut:

- Default akun

Anda dapat mengatur IMDSv2 agar diperlukan sebagai default akun. Ketika sebuah instance diluncurkan, konfigurasi instans secara otomatis diatur ke default akun.

Untuk mengatur default akun, lakukan hal berikut:

- Konsol Amazon EC2: Di Dasbor EC2, di bawah atribut Akun, Perlindungan dan keamanan data, untuk default IMDS, setel layanan metadata Instans ke versi Aktif dan Metadata hanya ke V2 (diperlukan token). Untuk informasi selengkapnya, lihat [Tetapkan IMDSv2 sebagai default untuk akun](#).
- AWS CLI: Gunakan perintah CLI [modify-instance-metadata-defaults](#) dan tentukan `dan. --http-tokens required --http-put-response-hop-limit 2`

- Instans baru

Saat meluncurkan instans baru, Anda dapat melakukan hal berikut:

- Konsol Amazon EC2: Di wizard instans peluncuran, atur Metadata yang dapat diakses ke Aktif dan Versi metadata ke V2 saja (diperlukan token). Untuk informasi selengkapnya, lihat [Konfigurasi instans saat peluncuran](#).
- AWS CLI: Gunakan perintah CLI [run-instance](#) dan tentukan bahwa IMDSv2 diperlukan.

- Instans yang ada

Untuk instans yang ada, Anda dapat melakukan hal berikut:

- Konsol Amazon EC2: Pada halaman Instans, pilih instans Anda, pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans, dan untuk IMDSv2, pilih Wajib. Untuk informasi selengkapnya, lihat [Mengharuskan penggunaan IMDSv2](#).

- AWS CLI: Gunakan perintah CLI [modify-instance-metadata-options](#) untuk menetapkan bahwa hanya IMDSv2 yang akan digunakan.

Anda dapat memodifikasi opsi metadata instans pada instans yang sedang berjalan, dan Anda tidak perlu memulai ulang instans setelah memodifikasi opsi metadata instans.

Langkah 4: Periksa apakah instans Anda dialihkan ke IMDSv2

Anda dapat memeriksa apakah ada instans yang belum dikonfigurasi untuk mengharuskan penggunaan IMDSv2, dengan kata lain, IMDSv2 masih dikonfigurasi sebagai `optional`. Jika ada instans yang masih dikonfigurasi sebagai `optional`, Anda dapat memodifikasi opsi metadata instans untuk membuat IMDSv2 `required` dengan mengulangi [Langkah 3](#) sebelumnya.

Untuk memfilter instans Anda:

- Konsol Amazon EC2: Pada halaman Instans, filter instans Anda dengan menggunakan filter IMDSv2 = opsional. Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter sumber daya menggunakan konsol](#). Anda juga dapat melihat apakah IMDSv2 diharuskan atau opsional untuk setiap instans: Di jendela Preferensi, aktifkan tombol IMDSv2 untuk menambahkan kolom IMDSv2 ke tabel Instans.
- AWS CLI: Gunakan perintah CLI [describe-instances](#) 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
```

Langkah 5: Jika semua instans Anda dialihkan ke IMDSv2

Kunci kondisi `ec2:MetadataHttpTokensec2:MetadataHttpPutResponseHopLimit`, dan `ec2:MetadataHttpEndpoint` IAM dapat digunakan untuk mengontrol penggunaan [ModifyInstanceMetadataOptions](#) API [RunInstances](#) dan 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](#)

Selanjutnya, setelah menonaktifkan IMDSv1, Anda dapat menggunakan MetadataNoTokenRejected CloudWatch metrik untuk melacak berapa kali panggilan IMDSv1 dicoba dan ditolak. Jika, setelah menonaktifkan IMDSv1, Anda memiliki perangkat lunak yang tidak berfungsi dengan baik dan MetadataNoTokenRejected metrik mencatat panggilan IMDSv1, kemungkinan perangkat lunak ini perlu diperbarui untuk menggunakan IMDSv2.

Menggunakan AWS SDK yang didukung

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 for .NET](#) – 3.3.634.1
- [AWS SDK for C++](#) – 1.7.229
- [AWS SDK for Go](#) – 1.25.38
- [AWS SDK for Go v2](#) - 0.19.0
- [AWS SDK for 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 for PHP](#) – 3.147.7
- [AWS SDK untuk Python \(Botocore\)](#) - 1.13.25
- [AWS SDK for Python \(Boto3\)](#) – 1.12.6

- [AWS SDK for Ruby](#) – 3.79.0

Mengonfigurasi opsi metadata instans

Layanan metadata instance (IMDS) berjalan secara lokal pada setiap instans EC2. Opsi metadata instance 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 sebuah instance. Saat dinonaktifkan, Anda atau kode apa pun tidak akan dapat mengakses metadata instance 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 diaktifkan secara otomatis. Jika Anda ingin mengaktifkan titik akhir IPv6, Anda perlu melakukannya secara eksplisit.

Titik akhir IMDS IPv6: | 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 [instans yang dibangun di Sistem AWS Nitro dan dalam subnet yang didukung IPv6 \(hanya tumpukan ganda atau IPv6\)](#).

Versi metadata: | IMDSv1 or IMDSv2 (token optional) IMDSv2 only (token required)

Saat meminta metadata instance, 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 hanya mengizinkan panggilan IMDSv2 (di mana token diperlukan).

Batas hop respons metadata: — 1 64

Batas hop adalah jumlah hop jaringan yang diizinkan untuk dilakukan oleh respons PUT. Anda dapat mengatur batas hop ke minimum 1 dan maksimum 64. Di lingkungan kontainer, kami sarankan untuk mengatur batas hop ke 2. Untuk informasi selengkapnya, lihat [Pertimbangan](#).

Akses ke tag dalam metadata contoh: | enabled disabled

Anda dapat mengaktifkan atau menonaktifkan akses ke tag instans dari metadata instans. Untuk informasi selengkapnya, lihat [Bekerja dengan tanda instans dalam metadata instans](#).

Tempat mengkonfigurasi 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. Nilai default tingkat akun tidak memengaruhi instance yang ada.
- AMI - Anda dapat mengatur `imds-support` parameter `v2.0` saat Anda mendaftar atau memodifikasi AMI. Saat instance diluncurkan dengan AMI ini, versi metadata instance secara otomatis disetel ke IMDSv2 dan batas hop disetel ke 2.
- Instance - Anda dapat mengubah semua opsi metadata instance pada instance saat peluncuran, mengesampingkan pengaturan default. Anda juga dapat mengubah opsi metadata instans 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 pada 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 dalam template peluncuran atau dalam konfigurasi instance. Nilai apa pun yang ditentukan di sini mengesampingkan nilai yang ditentukan di tingkat akun atau di AMI.
- Prioritas 2: Pengaturan akun - Jika nilai tidak ditentukan pada peluncuran instance, maka itu ditentukan oleh pengaturan tingkat akun (yang ditetapkan untuk masing-masing). Wilayah AWS Pengaturan tingkat akun menyertakan nilai untuk setiap opsi metadata, atau menunjukkan tidak ada preferensi sama sekali.
- Prioritas 3: Konfigurasi AMI — Jika nilai tidak ditentukan pada peluncuran instance atau pada tingkat akun, maka nilai tersebut ditentukan oleh konfigurasi AMI. Ini hanya berlaku untuk `HttpTokens` dan `HttpPutResponseHopLimit`.

Setiap opsi metadata dievaluasi secara terpisah. Instance 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 instance yang berjalan atau dihentikan, kecuali perubahan dibatasi oleh kebijakan IAM atau SCP.

Menentukan nilai untuk opsi metadata — Contoh 1

Dalam contoh ini, instans EC2 diluncurkan ke Wilayah di mana `HttpPutResponseHopLimit` diatur ke 1 tingkat akun. AMI yang ditentukan telah `ImdsSupport` disetel ke `v2.0`. Tidak ada opsi metadata yang ditentukan langsung pada instance saat peluncuran. Instans 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 templat peluncuran.
- Pengaturan akun diutamakan berikutnya: Dengan tidak adanya nilai spesifik yang ditentukan saat peluncuran, pengaturan di tingkat akun dalam Wilayah diutamakan. Ini berarti bahwa nilai default yang dikonfigurasi pada tingkat akun diterapkan. Dalam hal ini, `HttpPutResponseHopLimit` diatur 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 AMI diganti oleh pengaturan tingkat akun `HttpPutResponseHopLimit: 1`, yang memiliki prioritas lebih tinggi.

Menentukan nilai untuk opsi metadata — Contoh 2

Dalam contoh ini, instans EC2 diluncurkan dengan pengaturan yang sama seperti pada Contoh 1 sebelumnya, tetapi dengan `HttpTokens` disetel ke `optional` langsung pada instance saat peluncuran. Instans 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, `HttpTokens` disetel ke `required`), nilai yang ditentukan pada instance saat peluncuran (`HttpTokens` disetel ke `optional`) diutamakan.

Mengatur versi metadata instance

Ketika sebuah instance diluncurkan, nilai untuk versi metadata instance adalah salah satu atau `IMDSv1 or IMDSv2 (token optional)`. `IMDSv2 only (token required)`

Saat peluncuran instance, Anda dapat menentukan nilai untuk versi metadata secara manual, atau menggunakan nilai default. Jika Anda menentukan nilainya secara manual, itu akan mengganti default apa pun. Jika Anda memilih untuk tidak menentukan nilai secara manual, itu akan ditentukan oleh kombinasi pengaturan default, seperti yang diuraikan dalam tabel berikut.

Tabel menunjukkan bagaimana versi metadata untuk sebuah instance saat peluncuran (ditunjukkan oleh konfigurasi instans yang dihasilkan di kolom 4) ditentukan oleh pengaturan pada tingkat konfigurasi yang berbeda. Urutan prioritas adalah dari kiri ke kanan, di mana kolom pertama diutamakan, sebagai berikut:

- Kolom 1: Parameter peluncuran - Merupakan pengaturan pada instance yang Anda tentukan secara manual saat peluncuran.
- Kolom 2: Tingkat akun default - Merupakan pengaturan untuk akun.
- Kolom 3: AMI default - Merupakan pengaturan pada AMI.

Parameter peluncuran	Tingkat akun default	AMI standar	Konfigurasi instance yang dihasilkan
Hanya V2 (token diperlukan)	Tidak ada preferensi	Hanya V2	Hanya V2

Parameter peluncuran	Tingkat akun default	AMI standar	Konfigurasi instance yang dihasilkan
Hanya V2 (token diperlukan)	Hanya V2	Hanya V2	Hanya V2
Hanya V2 (token diperlukan)	V1 atau V2	Hanya V2	Hanya V2
V1 atau V2 (token opsional)	Tidak ada preferensi	Hanya V2	V1 atau V2
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 diatur	Tidak ada preferensi	Hanya V2	Hanya V2
Tidak diatur	Hanya V2	Hanya V2	Hanya V2
Tidak diatur	V1 atau V2	Hanya V2	V1 atau V2
Hanya V2 (token diperlukan)	Tidak ada preferensi	null	Hanya V2
Hanya V2 (token diperlukan)	Hanya V2	null	Hanya V2
Hanya V2 (token diperlukan)	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

Parameter peluncuran	Tingkat akun default	AMI standar	Konfigurasi instance yang dihasilkan
V1 atau V2 (token opsional)	V1 atau V2	null	V1 atau V2
Tidak diatur	Tidak ada preferensi	null	V1 atau V2
Tidak diatur	Hanya V2	null	Hanya V2
Tidak diatur	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 meminta penggunaan IMDSv2.

Mengonfigurasi opsi metadata instans untuk instans baru

Anda dapat mengonfigurasi opsi metadata instance berikut untuk instance baru.

Opsi

- [Mengharuskan penggunaan IMDSv2](#)
- [Aktifkan titik akhir IMDS IPv4 dan IPv6](#)
- [Nonaktifkan akses untuk metadata instans](#)

Mengharuskan penggunaan IMDSv2

Anda dapat menggunakan metode berikut untuk meminta penggunaan IMDSv2 pada instance baru. Anda.

Untuk mengharuskan IMDSv2

- [Tetapkan IMDSv2 sebagai default untuk akun](#)
- [Konfigurasi instans saat peluncuran](#)
- [Konfigurasi AMI](#)
- [Gunakan kebijakan IAM](#)

Tetapkan 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 instance yang ada. Misalnya, jika Anda menyetel default tingkat akun ke IMDSv2, semua instance yang ada yang disetel ke IMDSv1.

tidak akan terpengaruh. Jika Anda ingin mengubah nilai pada instance yang ada, Anda harus secara manual mengubah nilai pada instance itu sendiri.

Anda dapat mengatur default akun untuk versi metadata instans ke IMDSv2 sehingga semua instance 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` diatur ke `required` dan `HttpPutResponseHopLimit` diatur ke 2.

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 EC2.
4. Di bawah Atribut akun, pilih Perlindungan dan keamanan data.
5. Di samping default IMDS, pilih Kelola.
6. Pada halaman Kelola default IMDS, lakukan hal berikut:
 - a. Untuk layanan metadata Instance, pilih Diaktifkan.
 - b. Untuk Versi metadata, pilih V2 saja (token diperlukan).
 - c. Untuk batas hop respons Metadata, tentukan 2 jika instance Anda akan meng-host container. Jika tidak, pilih Tidak ada preferensi. Ketika tidak ada preferensi yang

ditentukan, saat peluncuran, nilai default ke 2 jika AMI memerlukan ImDSv2; jika tidak maka defaultnya menjadi 1.

- d. Pilih Perbarui.

AWS CLI

Untuk mengatur IMDSv2 sebagai default untuk akun untuk Wilayah yang ditentukan

Gunakan perintah [modify-instance-metadata-defaults](#) dan tentukan Region untuk memodifikasi pengaturan level akun IMDS. Sertakan `--http-tokens` set ke `required` dan `--http-put-response-hop-limit` atur ke 2 jika instance Anda akan meng-host kontainer. Jika tidak, tentukan `-1` untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default ke 2 jika AMI memerlukan ImDSv2; jika tidak maka defaultnya. 1

```
aws ec2 modify-instance-metadata-defaults \  
  --region us-east-1 \  
  --http-tokens required \  
  --http-put-response-hop-limit 2
```

Output yang diharapkan

```
{  
  "Return": true  
}
```

Untuk melihat pengaturan akun default untuk opsi metadata instance untuk Wilayah yang ditentukan

Gunakan [get-instance-metadata-defaults](#) perintah dan tentukan Wilayah.

```
aws ec2 get-instance-metadata-defaults --region us-east-1
```

Contoh Output

```
{  
  "AccountLevel": {  
    "HttpTokens": "required",  
    "HttpPutResponseHopLimit": 2  
  }  
}
```

```
}
```

Konfigurasi 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 untuk menyetel batas hop ke 2 Untuk informasi selengkapnya, lihat [Pertimbangan](#).

New 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](#).

Old console

Untuk mengharuskan penggunaan IMDSv2 pada instans baru

- Saat meluncurkan instans baru di konsol Amazon EC2, pilih opsi berikut di halaman Konfigurasi Detail Instans:
 - Pada Detail Tingkat Lanjut, untuk Metadata yang dapat diakses, pilih Diaktifkan.

- Untuk Versi metadata, pilih V2 (token diperlukan).

Untuk informasi selengkapnya, lihat [Langkah 3: Konfigurasi Detail Instans](#).

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 untuk menyetel 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 to `enabled` dan parameter ke. `MetadataOptions_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.

```
New-EC2Instance `\  
  -ImageId ami-0abcdef1234567890 `\  
  -InstanceType c6i.large `\  
  -MetadataOptions_HttpEndpoint enabled `\  
  -MetadataOptions_HttpTokens required
```

AWS CloudFormation

Untuk menentukan opsi metadata untuk instance yang digunakan AWS CloudFormation, lihat [AWS::EC2::LaunchTemplate MetadataOptions](#) properti di AWS CloudFormation Panduan Pengguna.

Konfigurasi 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 diatur ke V2 saja (token diperlukan) (konsol) atau `HttpTokens` diatur 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 mengonfigurasi 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 \  
  --imds-support v2.0
```

```
--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.

```
Import-Module AWS.Tools.EC2 # Required for Amazon.EC2.Model object creation.
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-0123456789example \  
--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 juga dapat membuat kebijakan IAM yang mencegah pengguna meluncurkan instans baru, kecuali mereka mengharuskan IMDSv2 pada instans baru.

Untuk menegakkan penggunaan IMDSv2 pada semua instans baru dengan menggunakan kebijakan IAM

Untuk memastikan bahwa pengguna IAM hanya dapat meluncurkan instans yang mengharuskan penggunaan IMDSv2 saat meminta metadata instans, Anda dapat menentukan bahwa kondisi yang mengharuskan IMDSv2 harus dipenuhi sebelum instans dapat diluncurkan. Untuk contoh kebijakan IAM, lihat [Cara menggunakan metadata instans](#).

Aktifkan titik akhir IMDS IPv4 dan IPv6

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 diaktifkan secara otomatis. Titik akhir IPv6 tetap dinonaktifkan bahkan jika Anda meluncurkan instance ke subnet khusus IPv6. Untuk mengaktifkan titik akhir IPv6, Anda perlu melakukannya secara eksplisit. Saat Anda mengaktifkan titik akhir IPv6, titik akhir IPv4 tetap diaktifkan.

Anda dapat mengaktifkan titik akhir IPv6 saat peluncuran instans atau setelahnya.

Persyaratan untuk mengaktifkan titik akhir IPv6

- Jenis instance yang dipilih dibangun di atas [Sistem AWS Nitro](#).

- Subnet yang dipilih mendukung IPv6, di mana subnet adalah [dual stack](#) atau IPv6 saja.

Gunakan salah satu metode berikut untuk meluncurkan instance dengan titik akhir IMDS IPv6 diaktifkan.

New console

Untuk mengaktifkan titik akhir IMDS IPv6 saat peluncuran instans

- [Luncurkan instans](#) di konsol Amazon EC2 dengan menentukan hal-hal berikut ini pada Detail lanjutan:
 - Untuk titik akhir Metadata IPv6, pilih Diaktifkan.

Untuk informasi selengkapnya, lihat [Detail lanjutan](#).

AWS CLI

Untuk mengaktifkan titik akhir IMDS IPv6 saat peluncuran instans

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 IMDS IPv6 saat peluncuran instans

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.

New 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](#).

Old console

Untuk menonaktifkan akses ke metadata instans saat peluncuran

- Luncurkan instans di konsol Amazon EC2 dengan pilih opsi berikut di halaman Konfigurasi Detail Instans:
 - Pada Detail Tingkat Lanjut, untuk Metadata yang dapat diakses, pilih Dinonaktifkan.

Untuk informasi selengkapnya, lihat [Langkah 3: Konfigurasi Detail Instans](#).

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 ke `disabled`

```
New-EC2Instance `\  
  -ImageId ami-0abcdef1234567890 `\  
  -InstanceType c6i.large `\  
  -MetadataOptions_HttpEndpoint disabled
```

AWS CloudFormation

Untuk menentukan opsi metadata untuk instance yang digunakan AWS CloudFormation, lihat [AWS::EC2::LaunchTemplate MetadataOptions](#) properti di AWS CloudFormation Panduan Pengguna.

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 memodifikasi opsi metadata instance, tentukan kebijakan yang mencegah semua pengguna selain pengguna dengan peran tertentu untuk menggunakan API. [ModifyInstanceMetadataOptions](#) Untuk contoh kebijakan IAM, lihat [Cara menggunakan metadata instans](#).

Mengueri opsi metadata instans untuk instans yang ada

Anda dapat mengueri opsi metadata instans untuk instans Anda yang ada dengan menggunakan salah satu metode berikut.

Console

Untuk mengueri opsi metadata instans untuk instans yang sudah ada menggunakan konsol

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. Tinjau opsi metadata instans saat ini di kotak dialog Ubah opsi metadata instans.

AWS CLI

Untuk menanyakan opsi metadata instance untuk instance yang ada menggunakan AWS CLI

Gunakan perintah CLI [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-EC2InstanceCmdlet](#).

```
(Get-EC2Instance `
  -InstanceId i-1234567898abcdef0).Instances.MetadataOptions
```

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. MetadataNoTokenCloudWatch Metrik melacak panggilan IMDSv1. Ketika MetadataNoToken mencatat nol penggunaan IMDSv1 untuk sebuah instance, instance kemudian siap untuk membutuhkan IMDSv2.

Console

Untuk mengharuskan penggunaan IMDSv2 pada instans yang 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 yang 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-1234567898abcdef0 \  
  --http-tokens required \  
  --http-endpoint enabled
```

PowerShell

Untuk mengharuskan penggunaan IMDSv2 pada instans yang ada

Gunakan [Edit-EC2InstanceMetadataOption](#) Cmdlet dan atur `HttpTokens` parameternya ke `required`. Jika Anda menetapkan nilai untuk `HttpTokens`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
(Edit-EC2InstanceMetadataOption \  
  -InstanceId i-1234567898abcdef0 \  
  -HttpTokens required \  
  -HttpEndpoint enabled).InstanceMetadataOptions
```

Memulihkan penggunaan IMDSv1

Jika IMDSv2 diharuskan, IMDSv1 tidak akan berfungsi saat meminta metadata instans. Ketika IMDSv2 bersifat opsional, maka IMDSv2 dan IMDSv1 akan berfungsi. Oleh karena itu, untuk memulihkan IMDSv1, jadikan IMDSv2 opsional dengan menggunakan salah satu metode berikut.

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-1234567898abcdef0 \  
  --http-tokens optional \  
  --http-endpoint enabled
```

PowerShell

Untuk memulihkan penggunaan IMDSv1 pada instans

Anda dapat menggunakan [Edit-EC2InstanceMetadataOption](#) Cmdlet dengan `HttpTokens` set `optional` to mengembalikan penggunaan IMDSv1 saat meminta metadata instance.

```
(Edit-EC2InstanceMetadataOption \  
  -InstanceId i-1234567898abcdef0 \  
  -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 SDK AWS CLI dan yang mendukung perubahan 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-1234567898abcdef0 \  
  --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-1234567898abcdef0 \  
  -HttpPutResponseHopLimit 3 \  
  -HttpEndpoint enabled).InstanceMetadataOptions
```

Aktifkan titik akhir IMDS IPv4 dan IPv6

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 diaktifkan secara otomatis. Titik akhir IPv6 tetap dinonaktifkan bahkan jika Anda meluncurkan instance ke subnet khusus IPv6. Untuk mengaktifkan titik akhir IPv6, Anda perlu melakukannya secara eksplisit. Saat Anda mengaktifkan titik akhir IPv6, titik akhir IPv4 tetap diaktifkan.

Anda dapat mengaktifkan titik akhir IPv6 saat peluncuran instans atau setelahnya.

Persyaratan untuk mengaktifkan titik akhir IPv6

- Jenis instance yang dipilih dibangun di atas [Sistem AWS Nitro](#).
- Subnet yang dipilih mendukung IPv6, di mana subnet adalah [dual stack](#) atau IPv6 saja.

Saat ini hanya AWS CLI dan AWS SDK yang mendukung mengaktifkan titik akhir IMDS IPv6 setelah peluncuran instance.

AWS CLI

Untuk mengaktifkan titik akhir IMDS IPv6 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-1234567898abcdef0 \  
  --http-protocol-ipv6 enabled \  
  --http-endpoint enabled
```

PowerShell

Untuk mengaktifkan titik akhir IMDS IPv6 untuk instans Anda

Gunakan [Edit-EC2InstanceMetadataOption](#) Cmdlet dan atur `HttpProtocolIpv6` parameternya ke `enabled`. Perhatikan bahwa saat Anda menetapkan nilai untuk `HttpProtocolIpv6`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
(Edit-EC2InstanceMetadataOption \  
  -InstanceId i-1234567898abcdef0 \  
  -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 mengembalikan 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-1234567898abcdef0 \  
  --http-endpoint enabled
```

PowerShell

Untuk mengaktifkan akses ke metadata instans

Gunakan [Edit-EC2InstanceMetadataOption](#) Cmdlet dan atur `HttpEndpoint` parameter-nya ke `enabled`

```
(Edit-EC2InstanceMetadataOption \  
  -InstanceId i-1234567898abcdef0 \  
  -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 mengembalikan 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-1234567898abcdef0 \  
  --http-endpoint disabled
```

PowerShell

Untuk menonaktifkan akses untuk metadata instans

Gunakan [Edit-EC2InstanceMetadataOption](#) Cmdlet dan atur `HttpEndpoint` parameter-nya ke `disabled`

```
(Edit-EC2InstanceMetadataOption \  
  -InstanceId i-1234567898abcdef0 \  
  -HttpEndpoint disabled).InstanceMetadataOptions
```

Mengambil metadata instans

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.

Metadata instans dibagi menjadi beberapa kategori. Untuk deskripsi setiap kategori metadata instans, lihat [Kategori metadata instans](#).

Untuk melihat semua kategori metadata instans dari dalam instans yang sedang berjalan, gunakan URI IPv4 atau IPv6 berikut.

IPv4

```
http://169.254.169.254/latest/meta-data/
```

IPv6

```
http://[fd00:ec2::254]/latest/meta-data/
```

Alamat IP adalah alamat tautan lokal dan hanya valid dari instans. Untuk informasi selengkapnya, lihat [Alamat link-lokal](#) di panduan pengguna ini dan [Alamat tautan lokal](#) di Wikipedia.

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 [instans yang dibangun pada Sistem AWS Nitro](#) dan dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#).

Format perintah berbeda, bergantung pada apakah Anda menggunakan IMDSv1 atau IMDSv2. Secara default, Anda dapat menggunakan kedua versi IMDS. Untuk mengharuskan penggunaan IMDSv2, lihat [Gunakan IMDSv2](#).

Untuk mengambil metadata instance pada instance Linux

Anda dapat menggunakan alat seperti cURL, seperti yang ditunjukkan pada contoh berikut.

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/
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

Untuk mengambil metadata instance pada instance Windows

Anda dapat menggunakan PowerShell cmdlet untuk mengambil URI. Misalnya, jika Anda menjalankan versi 3.0 atau yang lebih baru PowerShell, gunakan cmdlet berikut.

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/meta-data/
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
```

Jika Anda tidak ingin menggunakannya PowerShell, Anda dapat menginstal alat pihak ketiga seperti GNU Wget atau cURL.

Important

Jika Anda menginstal alat pihak ketiga pada instans Windows, pastikan Anda membaca dokumentasi yang menyertainya dengan hati-hati, karena metode pemanggilan HTTP dan format output mungkin berbeda dari yang didokumentasikan di sini.

Biaya

Anda tidak dikenai biaya untuk permintaan HTTP yang digunakan untuk mengambil metadata instans dan data pengguna.

Pertimbangan

Untuk menghindari masalah dengan pengambilan metadata instans, pertimbangkan hal berikut:

- Di lingkungan kontainer, kami sarankan untuk mengatur batas hop ke 2.

AWS SDK menggunakan panggilan IMDSv2 secara default. Jika panggilan IMDSv2 tidak menerima respons, SDK akan mencoba lagi panggilan tersebut dan, jika masih gagal, akan menggunakan IMDSv1. Hal ini dapat mengakibatkan penundaan, terutama di lingkungan kontainer. Dalam lingkungan kontainer, jika batas hop adalah 1, respon IMDSv2 tidak kembali karena masuk ke kontainer dianggap sebagai hop jaringan tambahan. Untuk menghindari proses jatuh kembali ke IMDSv1 dan penundaan yang dihasilkan, kami sarankan Anda menetapkan hop limit ke 2 dalam lingkungan kontainer. Untuk informasi selengkapnya, lihat [Mengonfigurasi opsi metadata instans](#).

- (Hanya Windows) Buat AMI khusus dengan 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. Bentuk informasi lebih lanjut, lihat [Buat AMI dengan Windows Sysprep](#).

- Untuk IMDSv2, Anda harus menggunakan **/latest/api/token** saat mengambil token.

Menerbitkan permintaan PUT ke jalur khusus versi apa pun, misalnya `/2021-03-23/api/token`, akan menyebabkan layanan metadata menampilkan kesalahan 403 Forbidden. Perilaku ini memang disengaja.

- Jika IMDSv2 diperlukan, IMDSv1 tidak berfungsi.

Anda dapat memeriksa apakah IMDSv2 diperlukan untuk sebuah instans, sebagai berikut: Pilih instans untuk melihat detailnya, dan periksa nilai IMDSv2. Nilainya adalah Wajib (hanya IMDSv2 yang dapat digunakan) atau Opsional (IMDSv2 dan IMDSv1 dapat digunakan).

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).

Untuk permintaan yang dibuat menggunakan Layanan Metadata Instans Versi 2, kode kesalahan HTTP berikut dapat ditampilkan:

- 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.

Contoh pengambilan metadata instans

Contoh berikut memberikan perintah yang dapat Anda gunakan pada instans Amazon EC2. Format perintah berbeda untuk instance Linux dan Windows.

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.

Note

Untuk menghindari keharusan memperbarui kode Anda setiap kali Amazon EC2 merilis build metadata instans baru, sebaiknya gunakan `latest` di jalur, dan bukan nomor versi. Misalnya, gunakan `latest` sebagai berikut:

```
curl http://169.254.169.254/latest/meta-data/ami-id
```

Linux

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/  
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
```

IMDSv1

```
[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
```

Windows

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/
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
```

IMDSv1

```
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, lihat [Kategori metadata instans](#).

Linux

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/
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/
```

IMDSv1

```
[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/
```

Windows

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/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/
```

IMDSv1

```
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/
```

Dapatkan nilai untuk item metadata

Contoh-contoh ini mendapatkan nilai dari beberapa item metadata tingkat atas yang diperoleh pada contoh sebelumnya. Permintaan IMDSv2 menggunakan token yang disimpan yang dibuat di perintah contoh sebelumnya, dengan asumsi belum kedaluwarsa.

Linux

IMDSv2

```
[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
```

IMDSv1

```
[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
```

Windows

IMDSv2

```
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
```

IMDSv1

```
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.

Linux

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/public-keys/  
0=my-public-key
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/  
0=my-public-key
```

Windows

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/meta-data/public-keys/0=my-public-key
```

IMDSv1

```
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.

Linux

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/public-keys/0/
openssh-key
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/
openssh-key
```

Windows

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/meta-data/public-keys/0/openssh-key
openssh-key
```

IMDSv1

```
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).

Linux

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/public-keys/0/openssh-key
ssh-rsa MIICiTCCAFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMCMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDZVQHEwTZWf0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC0lBTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAd
BgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQHEwTZWf0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC0lBTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWxhZAdBgkqhkiG9w0BCQEWEG5vb251QGFT
YXpvbi5jb20wZGZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMak0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZncvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
```

```
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCcAaFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAAsTC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwax1Aoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

Windows

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/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCcAaFICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAAsTC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTC2XADZ4nB+BLyGVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
```

```
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
ssh-rsa MIICiTCcAfICCCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMAKGA1UEBhMC
VVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBA5TC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAd
BgkqhkiG9w0BCQEWEG5vb251QGftYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAKGA1UEBhMCMVVMxCzAJBgNVBAGTAldBMRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25z
b2x1MRIwEAYDVQQDEw1UZXR0Q21sYWx1eHAdBgkqhkiG9w0BCQEWEG5vb251QGft
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySwTc2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEI03IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITx0USQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb30hjZnzcvcQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp3780D8uTs7fLvJx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

Dapatkan ID subnet untuk instans

Contoh ini mendapatkan ID subnet untuk sebuah instans.

Linux

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/network/interfaces/mac/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/
mac/02:29:96:8f:6a:2d/subnet-id
```

```
subnet-be9b61d7
```

Windows

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/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id  
subnet-be9b61d7
```

IMDSv1

```
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

Dalam contoh berikut, instans contoh [mengaktifkan tanda pada metadata instans](#) dan tanda instans Name=MyInstance serta Environment=Dev.

Contoh ini mendapatkan semua kunci tanda instans untuk sebuah instans.

Linux

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/tags/instance  
Name  
Environment
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

Contoh berikut mendapat nilai kunci Name yang diperoleh pada contoh sebelumnya. Permintaan IMDSv2 menggunakan token yang disimpan yang dibuat di perintah contoh sebelumnya, dengan asumsi belum kedaluwarsa.

IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/tags/instance/Name
MyInstance
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

Windows

IMDSv2

```
PS C:\> $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/tags/instance
```

```
Name
Environment
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

Contoh berikut mendapat nilai kunci Name yang diperoleh pada contoh sebelumnya. Permintaan IMDSv2 menggunakan token yang disimpan yang dibuat di perintah contoh sebelumnya, dengan asumsi belum kedaluwarsa.

IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
GET -Uri http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/
instance/Name
MyInstance
```

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 kredensial AWS keamanan, hindari kueri kredensial selama setiap transaksi atau secara bersamaan dari sejumlah besar thread atau proses, karena hal ini dapat menyebabkan pembatasan. 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.

Batasi akses IMDS

Anda dapat mempertimbangkan untuk menggunakan aturan firewall lokal untuk menonaktifkan akses dari beberapa atau semua proses ke IMDS.

Note

[Untuk contoh yang dibangun di Sistem AWS Nitro, IMDS dapat dijangkau dari jaringan Anda sendiri ketika alat jaringan dalam VPC Anda, seperti router virtual, meneruskan paket ke](#)

[alamat IMDS, dan pemeriksaan sumber/tujuan default pada instance dinonaktifkan.](#) Untuk mencegah sumber dari luar VPC Anda mencapai IMDS, kami sarankan Anda memodifikasi konfigurasi alat jaringan untuk menjatuhkan paket dengan alamat IPv4 tujuan IMDS 169.254.169.254 dan, jika Anda mengaktifkan titik akhir IPv6, alamat IPv6 dari IMDS. [fd00:ec2::254]

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 menggunakan FreeBSD atau OpenBSD, 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.

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.

Bekerja dengan data pengguna instans

Anda dapat menggunakan data pengguna instans untuk menyesuaikan instans Anda. Saat meluncurkan instans, Anda dapat menyimpan parameter atau skrip sebagai data pengguna. Skrip apa pun dalam data pengguna dijalankan saat Anda meluncurkan instans. Anda dapat melihat data pengguna sebagai atribut instans. Anda juga dapat melihat data pengguna dari instans Anda melalui Layanan Metadata Instans (IMDS).

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 mengenkode base64 untuk Anda atau menerima input yang dienkode base64.
- Data pengguna dibatasi hingga 16 KB, dalam bentuk mentah, sebelum dienkode base64. Ukuran string dengan panjang n setelah encode base64 adalah $\text{ceil}(n/3)*4$.
- 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 instance Windows, Anda dapat mengonfigurasi pengaturan sehingga skrip data pengguna yang diperbarui dijalankan satu kali ketika Anda memulai instance atau setiap kali Anda reboot atau memulai instance.
- Data pengguna adalah atribut instans. Jika Anda membuat AMI dari instans, data pengguna instans tidak disertakan dalam AMI.

Tentukan data pengguna instans saat peluncuran

Anda dapat menentukan data pengguna saat meluncurkan sebuah instans. Untuk petunjuk konsol, lihat [Tentukan data pengguna instans saat peluncuran](#). Untuk contoh Linux yang menggunakan AWS CLI, lihat [the section called “Data pengguna dan AWS CLI”](#). Untuk contoh Windows yang menggunakan Alat untuk Windows PowerShell, lihat [the section called “Data pengguna dan Alat untuk Windows PowerShell”](#).

Ubah data pengguna instans

Anda dapat memodifikasi data pengguna untuk instans dengan volume root EBS. Instans harus berada dalam status berhenti. Untuk petunjuk konsol, lihat [Lihat dan perbarui data pengguna instans](#). Untuk contoh Linux yang menggunakan AWS CLI, lihat [modify-instance-attribute](#). Untuk contoh Windows yang menggunakan Alat untuk Windows PowerShell, lihat [the section called “Data pengguna dan Alat untuk Windows PowerShell”](#).

Ambil data pengguna instans dari instans 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 [instans yang dibangun pada Sistem AWS Nitro](#) dan dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#).

Untuk mengambil data pengguna dari sebuah instans, gunakan URI berikut.

```
http://169.254.169.254/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 ini mengembalikan data pengguna yang disediakan sebagai teks yang dipisahkan koma.

Linux

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/user-
data
1234, john, reboot, true | 4512, richard, | 173,,,
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/user-data
1234, john, reboot, true | 4512, richard, | 173,,,
```

Windows

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/user-data
1234, john, reboot, true | 4512, richard, | 173, , ,
```

IMDSv1

```
PS C:\> 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 ini mengembalikan data pengguna yang disediakan sebagai skrip.

Linux

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/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
```

```
chkconfig httpd on
```

Windows

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/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

```
PS C:\> 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>
```

Ambil data pengguna instans dari komputer Anda

Anda dapat mengambil data pengguna sebagai instans dari komputer Anda sendiri. Untuk petunjuk konsol, lihat [Lihat dan perbarui data pengguna instans](#). Untuk contoh yang menggunakan AWS CLI, lihat [Data pengguna dan AWS CLI](#). Untuk contoh yang menggunakan Alat untuk Windows PowerShell, lihat [Data pengguna dan Alat untuk Windows PowerShell](#).

Mengambil data dinamis

Untuk mengambil data dinamis dari dalam instans yang berjalan, gunakan URI berikut.

```
http://169.254.169.254/latest/dynamic/
```

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 [instans yang dibangun pada Sistem AWS Nitro](#) dan dalam [subnet yang didukung IPv6 \(dual stack atau IPv6\)](#) saja).

Contoh ini menunjukkan cara mengambil kategori identitas instans tingkat tinggi.

Linux

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
```

Windows

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](#).

Kategori metadata instans

Metadata instans dibagi menjadi beberapa kategori. Untuk mengambil metadata instans, tentukan kategori dalam permintaan, dan metadata akan ditampilkan 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 instans yang dibangun di atas [Sistem Nitro](#), Anda dapat mengambil metadata instans 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. Sebagai contoh, *mac* menunjukkan alamat MAC untuk antarmuka jaringan. Anda harus mengganti placeholder dengan nilai sebenarnya saat Anda mengambil metadata instans.

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>ami-id</code>	ID AMI yang digunakan untuk meluncurkan instans.	1.0
<code>ami-launch-index</code>	Jika Anda meluncurkan beberapa instance menggunakan <code>RunInstances</code> panggilan yang sama, nilai ini menunjukkan urutan peluncuran untuk setiap instance. Nilai instans pertama yang diluncurkan adalah 0. Jika Anda meluncurkan instans menggunakan Auto Scaling atau armada EC2, nilai ini selalu 0.	1.0
<code>ami-manifest-path</code>	Jalur ke file manifes AMI di Amazon S3. Jika Anda menggunakan AMI yang didukung Amazon EBS untuk meluncurkan instans, hasil yang dikembalikan adalah <code>unknown</code> .	1.0
<code>ancestor-ami-ids</code>	ID AMI dari setiap instans yang dibundel ulang untuk membuat AMI ini. Nilai ini hanya akan ada jika file manifes AMI berisi kunci <code>ancestor-amis</code> .	10/10/2007

Kategori	Deskripsi	Versi ketika kategori dirilis
autoscaling/target-lifecycle-state	<p>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.</p>	2021-07-15
block-device-mapping/ami	Perangkat virtual yang berisi sistem file root/boot.	15/12/2007
block-device-mapping/ebs N	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
elastic-gpus/assoc iations/ <i>elastic-gpu-id</i>	Jika ada GPU Elastis yang terpasang ke instans, berisi string JSON dengan informasi tentang GPU Elastis, termasuk ID dan informasi koneksinya.	30/11/2016

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>elastic-inference/associations/ <i>eia-id</i></code>	Jika ada akselerator Inferensi Elastis yang terpasang ke instans, berisi string JSON dengan informasi tentang akselerator Inferensi Elastis, termasuk ID dan tipenya.	29/11/2018
<code>events/maintenance/history</code>	Jika ada peristiwa pemeliharaan yang diselesaikan atau dibatalkan untuk instans tersebut, berisi string JSON dengan informasi tentang peristiwa tersebut. Untuk informasi selengkapnya, lihat Untuk melihat riwayat acara tentang peristiwa yang sudah selesai atau dibatalkan .	17/08/2018
<code>events/maintenance/scheduled</code>	Jika ada peristiwa pemeliharaan aktif untuk instans tersebut, berisi string JSON dengan informasi tentang peristiwa tersebut. Untuk informasi selengkapnya, lihat Melihat peristiwa terjadwal .	17/08/2018

Kategori	Deskripsi	Versi ketika kategori dirilis
events/recommendations/rebalance	<p>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.</p>	2020-10-27
hostname	<p>Jika instans EC2 menggunakan Penamaan berbasis IP (IPBN), ini adalah nama host DNS IPv4 privat dari instans tersebut. Jika instans EC2 menggunakan Penamaan berbasis sumber Daya (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 Tipe nama host instans Amazon EC2.</p>	1.0

Kategori	Deskripsi	Versi ketika kategori dirilis
iam/info	Jika ada peran IAM yang terkait dengan instance, berisi informasi tentang terakhir kali profil instans diperbarui, termasuk LastUpdated tanggal instans, InstanceProfileArn , dan InstanceProfile Id. Jika tidak, tidak ada.	12/01/2012
iam/security-credentials/role-name	Jika ada IAM role yang terkait dengan instans, <i>role-name</i> adalah nama peran, dan <i>role-name</i> berisi kredensial keamanan sementara yang terkait dengan peran tersebut (untuk informasi selengkapnya, lihat Mengambil kredensial keamanan dari metadata instans). Jika tidak, tidak ada.	12/01/2012
identity-credentials/ec2/info	Informasi tentang kredensial di identity-credentials/ec2/security-credentials/ec2-instance .	23/05/2018

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>identity-credentials/ec2/security-credentials/ec2-instance</code>	Kredensial untuk peran identitas instans yang memungkinkan perangkat lunak on-instance mengidentifikasi dirinya AWS untuk mendukung fitur seperti EC2 Instance Connect AWS Systems Manager dan Default Host Management Configuration. Kredensi ini tidak memiliki kebijakan yang dilampirkan, sehingga mereka tidak memiliki izin AWS API tambahan selain mengidentifikasi instance ke fitur tersebut. AWS Untuk informasi selengkapnya, lihat Peran identitas instans .	23/05/2018
<code>instance-action</code>	Memberi tahu instans bahwa instans harus di-boot ulang sebagai persiapan untuk pemaketan. Nilai yang valid: <code>none</code> <code>shutdown</code> <code>bundle-pending</code> .	01/09/2008
<code>instance-id</code>	ID instans ini.	1.0
<code>instance-life-cycle</code>	Opsi pembelian instans ini. Untuk informasi selengkapnya, lihat Opsi pembelian instans .	01/10/2019
<code>instance-type</code>	Tipe instans. Untuk informasi selengkapnya, lihat Jenis Instans Amazon EC2 .	29/08/2007

Kategori	Deskripsi	Versi ketika kategori dirilis
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
local-hostname	Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Jika instans EC2 menggunakan Penamaan berbasis IP (IPBN), ini adalah nama host DNS IPv4 privat dari instans tersebut. Jika instans EC2 menggunakan Penamaan berbasis sumber Daya (RBN), ini adalah RBN. Untuk informasi selengkapnya tentang penamaan IPBN, RBN, dan EC2, lihat Tipe nama host instans Amazon EC2 .	19/01/2007

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>local-ipv4</code>	Alamat IPv4 privat dari instans. Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Jika ini adalah instans khusus IPv6, item ini tidak diatur dan akan menghasilkan respons HTTP 404.	1.0
<code>mac</code>	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
<code>metrics/vhostmd</code>	Tidak lagi tersedia.	01/05/2011
<code>network/interfaces/macs/mac/device-number</code>	Nomor perangkat unik yang terkait dengan antarmuka itu. Nomor perangkat sesuai dengan nama perangkat; misalnya, <code>device-number</code> pada 2 adalah untuk perangkat eth2. Kategori ini sesuai dengan bidang <code>DeviceIndex</code> dan <code>device-index</code> yang digunakan oleh API Amazon EC2 dan perintah EC2 untuk AWS CLI.	01-01-2011
<code>network/interfaces/macs/mac/interface-id</code>	ID antarmuka jaringan.	01-01-2011
<code>network/interfaces/macs/mac/ipv4-associations/public-ip</code>	Alamat IPv4 yang terkait dengan setiap alamat IP publik dan ditetapkan ke antarmuka itu.	01-01-2011

Kategori	Deskripsi	Versi ketika kategori dirilis
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 privat dari instans. Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Jika ini adalah instans khusus IPv6, penamaannya berdasarkan sumber daya. Untuk informasi selengkapnya tentang IPBN dan RBN, lihat Tipe nama host instans Amazon EC2 .	19/01/2007
network/interfaces/macs/mac/local-ipv4s	Alamat IPv4 privat yang terkait dengan antarmuka. Jika ini adalah antarmuka jaringan khusus IPv6, 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/ <i>mac</i> /network-card	Indeks kartu jaringan. Beberapa tipe instans mendukung banyak kartu jaringan.	01/11/2020

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>network/interfaces/macs/mac/owner-id</code>	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
<code>network/interfaces/macs/mac/public-hostname</code>	DNS publik antarmuka (IPv4). Kategori ini hanya ditampilkan jika atribut <code>enableDnsHostnames</code> diatur ke <code>true</code> . 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
<code>network/interfaces/macs/mac/public-ipv4s</code>	Alamat IP publik atau alamat IP Elastis yang terkait dengan antarmuka. Mungkin ada beberapa alamat IPv4 pada sebuah instans.	01-01-2011
<code>network/interfaces/macs/mac/security-groups</code>	Grup keamanan yang memiliki antarmuka jaringan.	01-01-2011
<code>network/interfaces/macs/mac/security-group-ids</code>	ID grup keamanan yang memiliki antarmuka jaringan.	01-01-2011

Kategori	Deskripsi	Versi ketika kategori dirilis
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
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

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>placement/group-name</code>	Nama grup penempatan tempat instans diluncurkan.	24/08/2020
<code>placement/host-id</code>	ID host tempat instans diluncurkan. Hanya berlaku untuk Host Khusus.	24/08/2020
<code>placement/partition-number</code>	Jumlah partisi tempat instans diluncurkan.	24/08/2020
<code>placement/region</code>	AWS Wilayah di mana instance diluncurkan.	24/08/2020
<code>product-codes</code>	AWS Marketplace kode produk yang terkait dengan instance, jika ada.	01/03/2007
<code>public-hostname</code>	DNS publik instans (IPv4). Kategori ini hanya ditampilkan jika atribut <code>enableDnsHostnames</code> diatur ke <code>true</code> . 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
<code>public-ipv4</code>	Alamat IPv4 publik. Jika alamat IP Elastis dikaitkan dengan instans, nilai yang ditampilkan adalah alamat IP Elastis.	19/01/2007

Kategori	Deskripsi	Versi ketika kategori dirilis
<code>public-keys/0/openssh-key</code>	Kunci publik. Hanya tersedia jika disediakan pada waktu peluncuran instans.	1.0
<code>ramdisk-id</code>	ID dari RAM disk ditentukan pada waktu peluncuran, jika ada.	10/10/2007
<code>reservation-id</code>	ID reservasi.	1.0
<code>security-groups</code>	<p>Nama-nama grup keamanan yang diterapkan ke instans.</p> <p>Setelah peluncuran, Anda dapat mengubah grup keamanan instans. Perubahan tersebut tercermin di sini dan di <code>network/interfaces/mac/mac/security-groups</code>.</p>	1.0
<code>services/domain</code>	Domain untuk AWS sumber daya untuk Wilayah.	25/02/2014
<code>services/partition</code>	Partisi tempat sumber daya berada. Untuk AWS Wilayah standar, partisi adalah <code>aws</code> . Jika Anda memiliki sumber daya di partisi lain, maka partisi-nya adalah <code>aws-<i>partitionname</i></code> . Contohnya, partisi untuk sumber daya di Wilayah Tiongkok (Beijing) adalah <code>aws-cn</code> .	20/10/2015

Kategori	Deskripsi	Versi ketika kategori dirilis
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
spot/termination-time	Perkiraan waktu, dalam UTC, sistem operasi untuk Instans Spot Anda akan menerima sinyal shutdown. Item ini ada 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 termination-time .	05/11/2014
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 Mengizinkan akses ke tanda dalam metadata instans .	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 secara terperinci CloudWatch. Nilai yang valid: enabled disabled	04/04/2009
instance-identity/document	JSON berisi atribut instans, seperti instance-id, alamat IP privat, dll. Lihat Dokumen identitas instans .	04/04/2009
instance-identity/pkcs7	Digunakan untuk memverifikasi keaslian dokumen dan konten terhadap tanda tangan. Lihat Dokumen identitas instans .	04/04/2009
instance-identity/signature	Data yang dapat digunakan pihak lain untuk memverifikasi asal dan keasliannya. Lihat Dokumen identitas instans .	04/04/2009

Contoh Linux: Nilai indeks peluncuran AMI

Contoh ini menunjukkan bagaimana Anda dapat menggunakan data pengguna dan metadata instance untuk mengonfigurasi instance Linux 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 [instans yang dibangun pada Sistem AWS Nitro](#) dan dalam [subnet yang didukung IPv6 \(dual stack atau IPv6 saja\)](#).

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 instans memiliki metadata unik tertentu.

Instans 1

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

Instans 2

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

Instans 3

Metadata	Nilai
instance-id	i-0ee992212549ce0e7
ami-launch-index	2
public-hostname	ec2-67-202-51-225.compute-1.amazonaws.com

Metadata	Nilai
public-ipv4	67.202.51.225
local-hostname	ip-10-251-50-37.ec2.internal
local-ipv4	10.251.50.37

Instans 4

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
```

Dokumen identitas instans

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. Dokumen identitas instans diekspos (dalam format JSON plaintext) melalui Layanan Metadata Instans (IMDS). Alamat IPv4 169.254.169.254 adalah alamat link-local dan hanya valid dari instans. Untuk informasi selengkapnya, lihat [Alamat tautan-lokal](#) di Wikipedia. Alamat IPv6 [fd00:ec2::254] adalah alamat lokal yang unik dan hanya valid dari instans. Untuk informasi selengkapnya, lihat [Alamat lokal yang unik](#) di Wikipedia.

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 [Instans yang dibangun di atas Sistem AWS Nitro dan dalam subnet yang didukung IPv6 \(hanya tumpukan ganda atau IPv6\)](#).

Anda dapat mengambil dokumen identitas instans dari instans yang sedang berjalan kapan saja. Dokumen identitas instans mencakup informasi berikut:

Data	Deskripsi
accountId	ID AWS akun yang meluncurkan instance.
architecture	Arsitektur AMI yang digunakan untuk meluncurkan instans (i386 x86_64 arm64).
availabilityZone	Zona Ketersediaan tempat instans berjalan.
billingProducts	Produk penagihan instans.

Data	Deskripsi
devpayPro ductCodes	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.
marketpla ceProductCodes	Kode AWS Marketplace produk AMI digunakan untuk meluncurkan instance.
pendingTime	Tanggal dan waktu instans diluncurkan.
privateIp	Alamat IPv4 privat dari instans.
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 plaintext

Untuk mengambil dokumen identitas instans plaintext

Hubungkan ke instans dan jalankan perintah berikut.

Linux

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
```

IMDSv1

```
$ curl http://169.254.169.254/latest/dynamic/instance-identity/document
```

Windows

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:\> (Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/document).Content
```

IMDSv1

```
PS C:\> (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

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:

- Tanda tangan berenkode base64—Ini adalah hash SHA256 berenkode base64 dari dokumen identitas instans yang dienkripsi menggunakan pasangan kunci RSA.
- Tanda tangan PKCS7—Ini adalah hash SHA1 dari dokumen identitas instans yang dienkripsi menggunakan pasangan kunci DSA.
- Tanda tangan RSA-2048—Ini adalah hash SHA256 dari dokumen identitas instans 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.

Topik berikut ini memberikan langkah-langkah yang mendetail untuk memvalidasi dokumen identitas instans menggunakan setiap tanda tangan.

- [Menggunakan tanda tangan PKCS7 untuk memverifikasi dokumen identitas instans](#)
- [Gunakan tanda tangan dengan encode base64 untuk memverifikasi dokumen identitas instans](#)
- [Menggunakan tanda tangan RSA-2048 untuk memverifikasi dokumen identitas instans](#)

Menggunakan tanda tangan PKCS7 untuk memverifikasi dokumen identitas instans

Topik ini menjelaskan cara memverifikasi dokumen identitas instans menggunakan tanda tangan PKCS7 dan sertifikat publik AWS DSA.

Instans Linux

Untuk memverifikasi dokumen identitas instans menggunakan tanda tangan PKCS7 dan sertifikat publik DSA AWS

1. Hubungkan dengan instans.

- 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
```

- Temukan sertifikat publik DSA untuk Wilayah Anda di [AWS sertifikat publik](#) dan tambahkan konten ke file baru yang bernama `certificate`.
- 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 AWS Support.

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 instans menggunakan tanda tangan PKCS7 dan sertifikat publik DSA AWS

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'} 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/pkcs7).Content)
```


`argument(s): "Cannot find the original signer. Jika tanda tangan tidak dapat diverifikasi, hubungi AWS Support.`

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, kontak AWS Support.

Gunakan tanda tangan dengan encode base64 untuk memverifikasi dokumen identitas instans

Topik ini menjelaskan cara memverifikasi dokumen identitas instance menggunakan tanda tangan berencode base64 dan sertifikat publik RSA AWS .

Instans Linux

Untuk memvalidasi dokumen identitas instance menggunakan tanda tangan berencode base64 dan sertifikat publik RSA AWS

1. Hubungkan dengan instans.
2. Ambil tanda tangan berencode 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](#) 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, `fileVerification 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 AWS Support.

Instans Windows

Untuk memvalidasi dokumen identitas instance menggunakan tanda tangan berenkode base64 dan sertifikat publik RSA AWS

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](#) 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 AWS Support.

Menggunakan tanda tangan RSA-2048 untuk memverifikasi dokumen identitas instans

Topik ini menjelaskan cara memverifikasi dokumen identitas instance menggunakan tanda tangan RSA-2048 dan sertifikat publik RSA-2048. AWS

Instans Linux

Untuk memverifikasi dokumen identitas instance menggunakan tanda tangan RSA-2048 dan sertifikat publik RSA-2048 AWS

1. Hubungkan dengan instans.
2. Ambil tanda tangan RSA-2048 dari metadata instans dan tambahkan ke file bernama `rsa2048` beserta 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. Tambahkan sertifikat publik RSA-2048 untuk Wilayah Anda di [AWS sertifikat publik](#) 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 rsa2048 -inform PEM -certfile certificate -noverify |  
tee document
```

Jika tanda tangan valid, pesan `Verification successful` muncul. Jika tanda tangan tidak dapat diverifikasi, kontak AWS Support.

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 RSA-2048 dan sertifikat publik RSA-2048 AWS

1. Hubungkan dengan instans.
2. Ambil tanda tangan RSA-2048 dari metadata instans, ubah ke byte array, dan tambahkan ke variabel bernama `$Signature`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.


```
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 Support.

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, kontak AWS Support.

AWS sertifikat publik

Sertifikat AWS publik berikut dapat digunakan untuk memverifikasi isi dokumen identitas instans seperti yang dijelaskan dalam topik berikut:

- [Verifikasi menggunakan tanda tangan PKCS7](#)
- [Verifikasi menggunakan tanda tangan berencode base64](#)
- [Verifikasi menggunakan tanda tangan RSA-2048](#)

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 disandikan base6, gunakan sertifikat RSA. Jika Anda memverifikasi tanda tangan RSA-2048, gunakan sertifikat RSA-2048.

Perluas setiap Wilayah di bawah ini untuk melihat sertifikat khusus Wilayah.

AS Timur (Ohio) — us-east-2

DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3Igb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUUVJTc+h0U+8Gk3JlqsX438Dk5c58wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzU2VydmljZXMgTEEx
MB4XDTE0MDQyOTEzMTU0V0XDTI1MDQyODEzMTU0V0VowXDELMAkGA1UEBhMVCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVzU2VydmljZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUUVJTc+h0U
+8Gk3JlqsX438Dk5c58wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAyWJQaVNWJqW0R0T0xV0SoN1GLk9x9kKEuN67RN9CLin4dA97qa7M1r5W4P
FZ6vnh5Cj0hQBRXV9xJUeYSdqVItNAUfK/fEzDdjf1nUfP1Q30J49u6CV01NoJ9m
usvY9kWcV46dqn2bk2MyfTTgvmeqP8fiMRPxxnVRkSz1ldP5Fg==
-----END CERTIFICATE-----

```

```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM07oeX4xevdMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA2MTAx
MjU0MThaGA8yMTk1MTEeNDEyNTgxOFowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhRGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWVlU2Vydm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAv6kGMnRmFDLxBEqXzP4npl65000kmQ7w8YXQygSdmNIOscGSU5wfh9
mZdcvCxXcDxgALFsFqPvH8fqIE9ttI0fEfuZvH0s8wUsIdKr0Zz0MjSx3cik4tKET
ch0EKfMnzK0gDBavraCDeX1rUDU0Rg7HFqNA0ry3uqDmnqtk00XC9GenS3z/7ebJ
fIBEPAm5oYMFVpX6M6St77WdNE8wEU8SuerQughIMVx9kMB07imeVHBiELbMQ0N
lwSWRL/61fA02keGSTfSp/0m3u+lesf2VwVFhqIJs+JbsEscPx0kIRlzy8mGd/JV
ONb/DQpTedzUKLgXbw7Kt03HTG9iXQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU2CTGYE5fTjx7gQXzdZSGPEWAJY4wgY4GA1UdIwSBhjCBg4AU2CTG
YE5fTjx7gQXzdZSGPEWAJY6hYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExB
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAM07oeX4xevdMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANDqkIpVypR2PveqUsAKke1wKCOsuw1UmH9k
xX1/VRoHbrI/UznrXtPQ0PMmHA2LKSTedwsJuorUn3cFH6qNs8ixBDrl8pZwfk0Y
IBJcTFBbI1xBEFkZo03wczzo5+8vPQ60RVqAaYb+iCa1HFJpccC30vajfa4GRdNb
n6FYnluIcDbmpcQePoVQwX7W3o0YLB1QLN7fE6H1j4TBI5Fd030uKzmaifQlWLYt
DVxVCNDabp0r6Uozd5ASm4ihPPoEoKo7I1p0f0T6fZ41U2xWA4+HF/89UoygZSo7
K+cQ90xGxJ+gm1YbLFR5rbJ0LfjrgDAb2ogbFy8LzHo2ZtSe60M=
-----END CERTIFICATE-----
```

AS Timur (Virginia) — us-east-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkiG00AQMDFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjA2MTAxMjU0MThaGA8y
MTk1MTEeNDEyNTgxOFowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgTODAxMDUxMjU0M
ThaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDg
YDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQz
CCAbcwggEsBgqhkiG00AQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
-----END CERTIFICATE-----
```

```

hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3Igb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUE1y2NIKCU+Rg4uu4u32koG9QEYIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAQoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTE
MB4XDTE0MDQyOTEzMDZQwMVowXDTI1MDQyODEzMDZQwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAe
BgNVBAQoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdT
ZWf0dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUE1y2NIKC
U+Rg4uu4u32koG9QEYIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQA1xSmwcWnhT4uAeSinJuz+1BTcKhVSWb5jT8pYjQb8ZoZkXXRGb09mvYeU
Neq0Br27rvRanaQ/9LUQf72+SahDFuS4CMI8nowoytqbmwquqFr4dxA/SDADyRiF
ea1UoMuNHTY49J/1vPomqsVn7mugTp+TbjqCf0Jtpu0temHcFA==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALFpzEAVWaQZMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdTZWf0
dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUE1y2NIKC
ODU5MTJaGA8yMTk1MDExNzA4NTkxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAQoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4OCQAQ8AMIIB
CgKCAQEAjS2vqZu9mE0h0q+0bRpAbCuiapbZMFNqRg7kT1r7Cf+gDqXKpHPjsng
SfNz+JHQd8WPI+pmNs+q0Z2aTe23klmf2U52KH9/j1k8R1Ibap/yFibFTSedmegX

```

```
E5r447GbJRSHUmuIIfZTZ/oRlpuII05/Vz7S0j22tdkdY2ADp7caZkNhxSP915fk
2jJMTBU0zyXUS2rBU/u1NHbTTeePjcEkvzVYPahD30TeQ+/A+uWUu89bHSQ0JR8h
Um4cFApzZgN3aD5j2LrSMu2pctkQwf9CaWyVznqrsGYjY0Y66LuFzSCXwqSnFBfv
fFBAFsJcGy24G2DoMyYkF3MyZlu+rwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAD
BgNVHQ4EFgQUrynsPp4uqSECwy+Pi04qyJ8TWSkwyY4GA1UdIwSBhjCBg4AUryns
Pp4uqSECwy+Pi04qyJ8TWSmhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJALFpzEAVWaQZMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADW/s81XijwdP6NkEoH1m9XLrvK4YTqkNfR6
er/uRRgTx2QjFcmNrx+g87gAm11lz+D0crAZ5LbEhDMs+JtZYR3ty0HkDk6SJM85
haoJNAFF7EQ/zCp1EJRiKLLsC7bcDL/Eriv1swt78/BB4RnC9W9kSp/sxd5svJMg
N9a6FAPlpNRsWAnbP8JBLAP93oJzb1X2LQXgykTghMkQ07NaY5hg/H5o4dMPc1TK
1YGq1FUCH6A2vdrxmpKDLmTn5//5pujdD2MN0df6sZwtxwZ0os1jV4rDjm9Q3VpA
NWIsDEcp3GUB4pro0R+C7PNkY+VG0DitB0w09qBGosCBstwyEqY=
-----END CERTIFICATE-----
```

AS Barat (California Utara) — us-west-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEsBXN0aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEXdBbWF6b24gV2ViIFNlcnZpY2VzIEExMQ4IeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBXN0aW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFNl
cnZpY2VzIEExMQ4IcCAbcwggEsBgqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
1Ra2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8Wqd+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
```

```

MIIDITCCAoqgAwIBAgIUK2zmY9PUSTR7rc1k20wPYu4+g7wwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVlZjZXMgTEExDQYJ
KjEwMDEwXDE0MDQyOFE3MDI0M1oXDTI5MDQyODE3MDI0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVlZjZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWduUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWduUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWV0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUK2zmY9PU
STR7rc1k20wPYu4+g7wwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQA1Ng4QmN4n7iPh5CnadS0c0ZfM7by0dBePwZJyGvOHdAw6P6E/vEk76KsC
Q8p+akuzVzVPkU4kBK/TRqLp19wEwoVwhhTaxHjQ1tTRHqXIVlRkw4JrtFbeNM21
G1kSLonuzmNZdivn9WuQYeGe7nUD4w3q9GgiF3CPorJe+UxtbA==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJANNPkIpcyEtIMA0GCSqGSIb3DQEBGwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEwMjkw
OTAzMDdaGA8yMTk1MDQwMzA5MMDMwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVlZjZXMgTEExDQYJKoZIhvcNAQELBQADggEBAFAA0CAQ8AMIIB
CgKCAQEAPhQgVhVq3SVcZDrC7575BW7GWLzCj8CLqYcL3YY7Jffupz70jcft057Z
4fo5Pj0CaS8DtPzh8+8vdwUSMbiJ6cDd3ooio3MnQ6DwzmsY+pY7CiI3UVG7KcH
4TriDqr1Iii7nB5MiPj8wTeAqX89T3SYaf6Vo+4Gcb3LCDGvnkZ9TrGcz2CHkJsJ
AIGwgopFpwhIjVYm7obmuIxSIUv+oNH0wXgDL029Zd98SnIYQd/njiqkzE+lvXgk
4h4Tu17xZIKBgFcTtWPKy+POGu81DYFqiWVEyR2JKKm2/iR1dL1YsT39kbNg47xY
aR129sS4nB5Vw3TRQA2jL0ToTIxzhQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUgepyi0Ns8j+q67dmcWu+mKKDa+gwgY4GA1UdIwSBhjCBg4AUgepy
i0Ns8j+q67dmcWu+mKKDa+ihYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExB
XYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IANNPkIpcyEtIMB1GA1UdEwEB/wQIMAYBAf8C
AAwDQYJKoZIhvcNAQELBQADggEBAFLWYutflU0xcAc+kmnMPqtC/Q6b79VIX0E
tNoKMI2KR81cV8ZE1XDb0NC6v8UeLpe1WBKjaWQtEjL1ifKg9hdY9Rj4RXIDSK7
33qCQ8juF4vep2U5TTBd6hfWxt1Izi88xudjixmbpUU4YK18UPbmixldYR+BEx0u
B1KJi9l1lxvuc/Igy/xeh0AZEjAXzVvHp8Bne33VVwMiMxWECZCiJxE4I7+Y6fqJ
pLLSFFJKbNaFyX1DiJ3kXyePEZSc1xiWeyRB2ZbTi5eu7vMG4i3AYWuFVLthaBgu

```

```
1PfHafJpj/JDcqt2vKUKfur5edQ6j1CGdxqqjawh0TEqcN8m7us=
-----END CERTIFICATE-----
```

AS Barat (Oregon) — us-west-2

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmveve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUfX8PxCKbHwpD31b0yCtyz3Gc1bgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWV2VydmljZXMgTEExDjE5
ODAxMDUxMjU2MTJaMB4XDTE0MDQy0TE3MjM10VowXDTI5MDQy0DE3MjM10VowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWV2VydmljZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwUB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXk3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUfX8PxCKb
```

```
HwpD31b0yCtyz3Gc1bgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBz0l+9Xy1+UsbUBI95H09mbbdnux+aMJXgG9uFZNjgNEbMcvx+h8P9IMko
z7PzFdheQQ1NLjsHH9mSR1SyC4m9ja6BsejH5nLBWyCdjfdP3muZM405+r7vUa10
dWU+hP/T7DUrPAIVM0E7mpYa+WPWJrN6B1RwQkKQ7twm9kDa1A==
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALZL31rQCSTMMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
OTAxMzJaGA8yMTk1MDEeNzA5MDEzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWV2VydmljZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA02Y59qtAA0a6uzo7nEQcnJ260KF+LRPwZfixBH+EbEN/Fx0gYy1jppCP
s5+VRNg6/WbfqAsV6X2VSjUKN59ZMnMY9ALA/Ipz0n00Huxj38EBZmX/NdNqKm7C
qWu1q5kmIvYjKGIadfbou8wLwLcHo8yvwfgI6FiGGsE09VMC56E/hL6Cohko11LW
dizyvRcvG/IidazVkJQCN/4zC9PU0VyKdhW33jXy8BTg/QH927QuNk+ZzD7HH//y
tIYxDhR6TIzSsnRjz3b0cEHxt1nsidc65mY0ejQty4hy7ioSiapw316mdbtE+RTN
fcH9FPIFKQNBpiqfAW5Ebp3La13/+wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU7coQx8Qnd75qA9XotSWT3IhvJmowgY4GA1UdIwSBhjCBg4AU7coQ
x8Qnd75qA9XotSWT3IhvJmqhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJALZL31rQCSTMMB1GA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFZ1e2MnzRaXCaLwEC1pW/f0oRG8nHr1PZ9W
OYZEWbh+QanRgaikBNDtVTwARQcZm3z+HWSkaIx3cyb6vM0DskZuiwzm1LJ9rDPc
aBm03SEt5v8mcc7sXWvgFjCnUpzomky6JheCD401Cf8k0o1Z93FQnTrbg620K0h
83mGCDeVKU3hLH97FYUq+3N/IliWFDhvbAYYKFJydZLhIdlCiiB99AM6Sg53rm
oukS3csyUxZyTU2hQfdjyo1nqW9yhvFAKjnnggiwxNKTPZzstKW8+cnYwiiTwJN
QpVoZdt0SfbuNnmwRUMi+QbuccXweav29QeQ3ADqjgB0CZdSRkk=
-----END CERTIFICATE-----
```

Afrika (Cape Town) — af-south-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7DCCAqwCCQCncbCtQbjuyzAJBgqhkiG9w0BAQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTA2MDQxMjQ4MDVaFw00
NTA2MDQxMjQ4MDVaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9u
```

```

IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbYwggErBgcqhkJ00AQBMIIBHgKBgQC12Nr1gMrHcFSZ7S/A
pQBSCMHWmn2qeoQTMVWqe50fnTd0zGFxDdIjKxUK58/8zjWG5uR4TXRzmZpGpmXB
bSufAR6BGqud2LnT/HIWGJAsnX2u0tSyNfCoJigqwha5w+CqZ6I7iBDdnB4TtTw
q06TlnExHFVj8LMky1ZgiaE1CQIVAIhdobse4K0QnbAhCL6R2euQz1oXAOgAV/21
WUuMz/79Ga0JvQcz1FNy1sT0pU9rU4TenqLQIt5iccn/7EIfNtvV05TZKulIKq7J
gXZr0x/KIT8zsNweetL0aGehPIYRMPX0vunMMR7hN7qA7W17WZv/76adywIsnDKq
ekfe15jinaX8MsKudyDK7Y+ifCG4PVhoM4+W2XwDgYQAAGAIx0KbVgwLxnb6Pi2
6hB0ihFv16jKxAQI0hHzXJLV0VYv9QwnqjJJRf0Cy3dB0zicLXiIxeIdYfvqJr+u
h1N8rGxEZYjYjEUKMGvsc0DW85jonXz0bNfcP0aaKH01KKVjL+0Zi5n2kn9wgd05
F3CVnM18BUra8A1Tr2yrrE6TVZ4wCQYHKoZiZjgEAWmVADAsAhQfa7MCJZ+/TEY5
AUr0J4wm8VzjoAIUSYZVu2NdRJ/ERPmDfhW5EsjH1CA=
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAKumfZiRrNvHMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTExMjcw
NzE0MDVaGA8yMTk5MDUwMjA3MTQwNVowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVjU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDFd571nUzVtke3rPyRkYfvs3jh0C0EMzzG72boyUNjnfW1+m0TeFraTLKb9T6F
7TuB/ZEN+vmlyqr2+5Va8U8qLbPF0bRH+FdaKjhgWZdYXxGzQzU3ioy5W5ZM1VyB
7iUsxEA1xsybC3ziPYaHI42UiTkQNaHmoroNeqVyHNnBpQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAAJLy1WylEgOpW4B1XPyRVD4pAds8Guw2+krqkY0HxLCdjosuH
RytGDGN+q75aAoXzW5a7SGpxLxk6Hfv0xp3RjDHsoeP0i1d8MD3hAC5ezxS4oukK
s5gbP0nokhKTMPXbTdRn5ZifCbWlx+bYN/mTYKvxho7b5SVg2o1La9aK
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAIIFI+05A6/ZIMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTA2MDQx
MjQ4MDRaGA8yMTk4MTEwNzEyNDgwNFowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVjU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQKCAQEAY7/WHBBH0rk+20aumT07g8rXrSM0UXgki3eYgKauPCG4Xx//vwQbuZwI
oeVmR9nqnfhij2w0cQdbLandh0EGtbxerete3IoXzd1KXJb11Pvmzrzyu5SPBPuP
iCeV4qdjjkXo2YWM6t9YQ911hcG96YSp89TBXFYUh3KLxfqAdTVhuC0NRGhXpyii

```

```
j/czo9njofHhqhTr7UEyPun8NVs2QWctLQ86N5zWR3Q0GRoVqqMrJs0cowHTrVw2
9Qr7QBjjB0VbyYmtYxm/DtiKprYV/e6bCAVok015X1sZDd3oC0QNoGlv5XbHJe2o
JFD8GRRy2rkW0/1NwVFDcweC6zC3QwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCE
goqzjpCpmMgCpszFHwvRaSMbspKtK7wNImUjrSB0fBjsfFuLyg1Zgn2nDCK7kQhx
jMjMNIvXbps3yMqQ2cHUKKcKf5t+WldfeT4Vvk1Rz6HSA8sd0kgVcIesIaoy2aaXU
VEB/oQziRGyKdN1d4TGYVZXG44CkrzSDv1bmfITq5tL+kAieznVF3bzHgPZW6hKP
EXC3G/IXrXicFEe6YyE1Rak162VncYSXiGe/i2XvsiNH3Q1mnx5XS7W0SCN0oAxW
EH9twibauv82DVg1W0kQu8EwFw8hFde9X0Rkiu0qVcuU81JgFEvPWMDFU5sGB6ZM
gkEKTzMv1ZpPbBhg99J1
-----END CERTIFICATE-----
```

Asia Pasifik (Hong Kong) – ap-east-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQC07MJe5Y3VLjAJBgqhkj00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTAyMDMwMjIxMjFaFw00
NTAyMDMwMjIxMjFaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbgwggEsBgcqhkj00AQDMIIBHwKBgQDvQ9RzVvf4MAwGbbqfX
b1CvCoVb99570kLGN/04CowHXJ+vTBR7eyIa6AoX1tsQXB0mrJswToFKKxT4gbuw
jK7s9QXX4CmTRWcEg02RXtZSVj0hsUQMh+yf7Ht40VL97LWnNfGsX2cwjCRWHYgI
71vnuBNBzLQHdSEwMNq0Bk76PwIVAMan6XIEEPnwr4e6u/RNnWBGkd9FAoGBAOCG
eSNmXPw4QFu4pI1Aykm6EnTZKKHT87gdXkAkfoC5fAf0xxhnE2HezZHp9Ap2tMV5
8bWNvoPHvoKCQqwfM+OUB1AxC/3vqoVkkL2mG1KgUH9+hrtPMtkw03RREnKe7I50
x9qDimJp0ihrl4I0dYvy9xU0oz+DzFAW8+y1WVYpA4GFAAKBgQDbnBAKSxWr9QHY
6Dt+EFdGz61AZLedeBKpaP53Z1DT034J0C55YbJTwBTFGqPtOLxnUVD1GiD6GbmC
80f3jvogPR1mSmGsydbNbZnbUEVWrRhe+y5zJ3g9qs/DWmDW0deEFvkhWVnLJkFJ
9pd0u/ibRPH11E2nz6pK7G60QtLyHTAJBgqhkj00AQDAzAAMC0CFQCoJlwGtJQC
cLoM4p/jtVF0j26xbgIUUS4pDKyHaG/eaygLtTfFJqzWHc=
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICszCCAbQCCQDtQvkVxRvK9TANBgqhkiG9w0BAQsFADBqMQswCQYDVQQGEwJV
UzETMBEGA1UECBMKV2FzaGluZ3RvbjEQMA4GA1UEBxMHU2VhdHRsZTEYMBYGA1UE
ChMPQW1hem9uLmNvbSBjbmuMR0wGAYDVQQDExF1YzIuYW1hem9uYXdzLmNvbTAe
Fw0xOTAyMDMwMzAwMDZaFw0yOTAyMDIwMzAwMDZaMGoxCzAJBgNVBAYTA1VTMRMw
EQYDVQQIEWpXYXNoaW5ndG9uMRAwDgYDVQQHEwdTZWF0dGx1MRgwFgYDVQKKEw9B
```

```

bWF6b24uY29tIEluYy4xGjAYBgNVBAMTEWVjMi5hbWF6b25hd3MuY29tMIGfMA0G
CSqSISb3DQEBAQUAA4GNADCBiQKBgQC1kkHXyTfc7gY5Q55JJhjTieHAgacaQkiR
Pity9QPDE3b+NXDh4UdP1xdIw73JcIIG3sG9RhWiXVCHh6KkuCTqJfPUknIKk8vs
M3RXf1UpBe8Pf+P92pxqPMCz1Fr2NehS3JhhpkCZVGxxwLC5gaG0Lr4rF0RubjYY
Rh84dK98VwIDAQAQMA0GCSqSISb3DQEBwUAA4GBAA6xV9f0HMqXjPHuGILDyaNN
dKcvp1NFwDTyVg32MNUbAGnecoEBtUPTxBsLoVYXC0b+b5/ZMDubPF9tU/vSXuo
TpYM5Bq57gJzDRaB0ntQbX9bgHiUxw6XZwaTS/6xjRjDT5p3S1E0mPI31P/eJv4o
Ezk5zb3eIf10/sqt4756
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAMoxixvs3YssMA0GCSqSISb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xODA3MjAw
ODQ0NDRAgA8yMTk3MTIyMzA4NDQ0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWVgU2Vydm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEA4T1PNs0g0FDrG1WePoHe0Sm0JTA3HCry5LSbYD33GFU2eBr0IxoU/+SM
rInKu3GghAMfH7WxPW3etIAZiyTDDU5RLcUq2Qwdr/ZpXAWpYocNc/CEmBFtfbxF
z4uwBIN3/drM0RSbe/wP9EcgmNUGQMMZWeAji8sMtwp0b1NWAP9BniUG0F1cz6Dp
uPovwDTLdAYT3Tyhz1ohKL3f6048TR5yTaV+3Ran2SGRhyJjfh3FRpP4VC+z5LnT
WPQHn74Kdq35UgrUxNhJraMGczzno1UuoR/tFMwR93401GsM9fVA7SW3jjCGF81z
PSzjy+ArKyQqIpLW1YGWDFk3sf08FQIDAQAQMA0GCSqSISb3DQEBwUAA4IBAQDK
2/+C3nPMgty0FX/I3Cyk+Pui44Ig0wCsIdNGwuJysdq5VIfnjegEu2zIMWJSKGO
1MzoQXjffkVZZ97J7RNDW06oB7kj3WVE8a7U4WE0fn0/CbMUF/x99CckNDwpjgW+
K8V8SzAsQDvYZs2KaE+18GFfLVF1TGUYK2rPSZMHyX+v/TI1c/qUceBycrIQ/kke
jDFsUMLqgm0V2hXKUpIsmiWMGrFQV4AeV0iXP8L/ZhcepL1t5SbsGdUA3AUy1
3If8s81uTheiQjwY5t9nM0SY/1Th/tL3+RaEI79VNEVfG1FQ8mgqCK0ar4m0oZJ1
tmmEJM7xeURdpBBx36Di
-----END CERTIFICATE-----

```

Asia Pasifik (Hyderabad) — ap-south-2

DSA

```

-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIJGAXjrQ4+XMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24g
U4EddRIPUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYldmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzriith1yrv8iIDGZ3RSAHHAUA12BQjxUjC8yykrmCouuEC/

```

```

BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBGLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1U1LZAFM0/7PSSoDgYUAAoGBAJCKGBoxIUxqBk94JHhwZZbgvbP0DA0oHENQWxp/981I7/
Y0fYJ0VMJS22aCnHDurofmo5rvNIkgXi7Rztbhu
+lko9rK6DgmpUwBU0WZtf34aZ2IWNBwHaVhHvWAQf9/46u18dMa2YucK1Wi+Vc+M
+K1drvGxmhym6ErN1zhJyMAkGByqGSM44BAMDlwAwLAIUaaPKxa0HoYvwz709xXpsQueIq+UCFFa/
GpzoD0Sok11057NU/2hnsiW4
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIICmzCCAZygAwIBAgIGAXjwLj9CMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
+sFcobrjvcAYm0PNRD8f4R1jAzvoLt2+qGe0TAY01Httj6cmsYN3AP1hN5iYuppFiYs12eNPa/
CD0Vg0BAfDF1V5rzjpA0j7TJabVh4kj7JvtD+xYMi6wEQA4x6SPONY40eZ2+8o/
HS8nucpWDVdPR06ciWULmHjmdmwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAAy6sgTdRkTqELHBeWj69q60xHyUmsWqHAQ
TGGbYP0yP2qfM10cCIImzRI5W0gn8gogderVfeT7nH5ih0TWEy/QDWfkQ601L4erm4yh4YQq8vcqAPSkf04N
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAIWfPw/X82fMA0GCSqGSIb3DQEBQwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA3MDQx
NDMwMjhaGA8yMjAxMTIwODE0MzAyOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlGdWUyVydmljZXMgTEExMjI0MjI0MjI0MjI0MjI0MjI0MjI0MjI0MjI0
CgKCAQEAQ29QEFriG+qFEjYw/v62nN701MJY/Hevx5TtmU/VIYBPQa3HUGTBAbbI
2Tmy8UMpa8kZeaYeI3RAfiQWt0Ws7wUrBu02Pdp518WDPaJUH7RWEuu1BDDkyZRW
NAMNPCn3ph70d243IFcLGku7HVeke15poqRpSfojrMasjlf+CvixUeAJbmFoxUHK
kh5unzG2sZy04wHXcJPQRf5a8zSTPe9YZP1kXPPEv4p/jTSggaYPxXyS6QVaT1V
zLeLFZ0fesLPMeil3KYQtV7IKLQiEA2F6dxWnxNWQ1yMHtdq6PucfEmVx17i/Xza
yNBRo0azY8WUNVKEEXrRhp/pU8Nh3GQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU9A01aZk9RLXk2ZvRVoUxYvQy9uwwgY4GA1UdIwSBhjCBg4AU9A01
aZk9RLXk2ZvRVoUxYvQy9uyhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAIVWfPw/X82fMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADEXluMRQRftqViahCnauEWGdMvLCBr8A+Yr
6hJq0guoxEk/lahxR137DnfMPuSbi1Rx5QKo7oBrWfG/zsgQUnF2IwHTzwD+i/2m
XCane6FiS5RpK31GdILq8ZmlhQk+6iI8yoZLr0LCfTh+CLgIKH0knfR51FzgzAiF
SI8/Q9mm+uvYtSTZECI6Zh57QZPoETAG/y1+9ji0y21Aelqa/k1i+Qo8gMf0c+Pm

```

```
dwY7o6fV+oucgr1sdey6VM45LeyILQqv0RXtVzjuowanzmCCFMjgqi09oZAWu40h
+F3unijELo01vZJs8s2N3KGlo3/jtUFTX6RTKShZ1APLwBi5GMI=
-----END CERTIFICATE-----
```

Asia Pasifik (Jakarta) — ap-southeast-3

DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXbvDEikMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrrmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAfEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utrZT
+ZxBxCBgLRJFnEj6EwoFh03zwykjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1U1LZAFM0/7PSSoDgYUAAoGBAPjuiEx05N3JQ6cVwntJie67D80uNo4jGRn
+crEtL7Y00jSVB9zGE1ga
+UgRPIaYETL293S8rTJTVgXAqdpBwfaHC6NUzre8U8iJ8FMNn1P9Gw1oUIlgQBj0RyynVJexoB31TDZM
+/52g90/bpq1QqNyKbeIgyBB1c1dAtr1QLnsMAkGByqGSM44BAMDlwAwLAIUK8E6RDIRtwK+9qnaTOBhv0/
njuQCFFocyT10xK+UDR888oNsdgtif2Sf
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICmzCCAZygAwIBAgIGAXbvVDG2yMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBYXNoaW5n
Vbt0gQ1ebWcur2hS07PnJifE40PxQ7RgSA1c4/spJp1sDP+ZrS0L01ZJfKhXf1R9S3AUwLnsC7b
+IuVXdY5LK9RKqu64nyXP5dx170zoL81oEyCSuRR2fs+04i2QsWBVP+KFNA7P5L1EHRjkT08kjNKvivrV
+0kP9ab5wIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAI4WUy6+DKh0JDSzQEZNYBgN1SoSuC2owtMxCwGB6nBfzzfcekWvs
+87w/g91NwUnUt0ZHYyh2tuBG6hVJuUEwDJ/z3wDd6wQviL0TF3MITawt9P8siR1hXqLJNxpjRQFZrgHqi
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMtdyRcH51j9MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWf6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA0MDgX
MjM5MTZaGA8yMjAxMDkxMjE5MzcxNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWV2VydmljZXMGTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
```

```
CgKCAQEAvUsKCxoh6KXRYJLeYTWAQfaBQeCwhJaR56mfUeFHJE4g8aFjWkiN4uc1
Tv0yYNNIZKTHWmzmulmdinWNbwP0GiR0Hb/i7ro0HhvnptyycGt8ag8affiIbx5X
7ohdwSN2KJ6G0IKf1Ix7f2NEI0oAMM/9k+T1eVF+MVWzpZoiDp8frLNkqp8+RAGz
ScZsbrfWv3u/if5xJAVdg2nckIWDMSHEVPoz01Jo7v0ZuDtwWsL1LHnL5ozvsKEk
+ZJyEi23r+U1hIT1NTBdp4yoigNQexedtwCSr7q36o0dDwvZpqY1kLi3uxZ4ta+a
01pz0STwMLgQZSbKWQrpMvsIAPrxoQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU1GgnGdNpbnL31LF30Jomg7Ji9hYwgY4GA1UdIwSBhjCBg4AU1Ggn
GdNpbnL31LF30Jomg7Ji9hahYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAMtdyRcH51j9MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACV100qQ1atBKVeiWMrhpczsJroxDx1ZT0ba
6wTMZk7c3akb6XM0SZFbGaiFkebPZqTHEhD1rC1M2j9A1LYcCx6YCrTf4cuhn2mD
gcJN33143e0WSaeRY3ee4j+V9ne98y3k02wLz95VrRgc1PFR8po2iWgZGhwUi+FG
q8dXeCH3N0DZgQsSqQWwmdNQXZZej6RHLU/8In5trHKLY0ppnLBjn/UZQbeTyW5q
RJB3GaveXjfgFUWj2q0cDuRGaikdS+dYaLsi5z9cA3FolHzWxx9M0s8io8vKqQzV
XUlrLTNwuhZy88c0lqGPxnoRbw7TmifwPw/cunNrsjUU0gs6ZTk=
-----END CERTIFICATE-----
```

Asia Pasifik (Melbourne) — ap-southeast-4

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjWF7P2MAkGByqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRipUt9Knc7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzriith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBGLRjFnEj6EwoFh03zwykjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1ULZAFM0/7PSSoDgYQAAoGAPRXSsQP9E3dw8QXK1rgBgEVCprLHdK/bbrMas0XMu1Eh0D
+q
+0PcTr8+iwbtoX1Y5MCeatWIp1GrXQjVqsF8vQqx1EuRuYKbR3nq4mWwaeG1x9AG5EjQHRa3GQ44wWH0dof0M3NRI1MP
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAzygAwIBAgIGAXjSh40SMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBXN0aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IJAMtdyRcH51j9MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACV100qQ1atBKVeiWMrhpczsJroxDx1ZT0ba
6wTMZk7c3akb6XM0SZFbGaiFkebPZqTHEhD1rC1M2j9A1LYcCx6YCrTf4cuhn2mD
gcJN33143e0WSaeRY3ee4j+V9ne98y3k02wLz95VrRgc1PFR8po2iWgZGhwUi+FG
q8dXeCH3N0DZgQsSqQWwmdNQXZZej6RHLU/8In5trHKLY0ppnLBjn/UZQbeTyW5q
RJB3GaveXjfgFUWj2q0cDuRGaikdS+dYaLsi5z9cA3FolHzWxx9M0s8io8vKqQzV
XUlrLTNwuhZy88c0lqGPxnoRbw7TmifwPw/cunNrsjUU0gs6ZTk=
-----END CERTIFICATE-----
```



```

cnZpY2VzIExMQzCCAbcwggEsBgcqhkJ00AQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
1Ra2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDlwAwLAIUwXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUDLA+x6tTAP3LRTTr0z6n0xfsozdMwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVudm1jZXMgTEExD
MB4XDTI0MDQyOTE0MTMwMVowXDTI0MDQyODE0MTMwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVudm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQChvRjF/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXk3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwCzAJ
BgNVBAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdT
ZWFOdGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIExMQzCCAbcwggEs
AP3LRTTr0z6n0xfsozdMwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAZ7rYKoAwwiiH1M5GJbrT/BEk3002VrEPw8ZxgpqQ/EK1zM10s/0Cyrmp7
UYyUgYfQe5nq37Z94r0USeMgv/WRxaMwrL1LqD78cuF9DSkXaZIX/kECtVaUnjk8
BZx0QhoIH0pQocJUS1m/dLeMuE0+0A3HNR6JVktGsUdv9u1mKw==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJAPRYyD8TtmC0MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIExMQzCCAbcwggEsAgFw0xNjAzMDcx
MDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft

```

```

YXpvbiBXZWVgU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEALSS5I/eCT2PM0+qusorBx67QL26BIWQHd/yF6ARtHbB/1DdFLRqE5Dj
07Xw7eENC+T79m0x0AbeWg91Ka0D0zw6i9I/2/HpK0+NDEdD6sPKDA1d45jRra+v
CqAjI+nV9Vw91wv7HjMk3RcjWGziM8/hw+3YNIutt7aQzZRwIw1Bpcqx3/AFd8Eu
2UsRMSHgkGUW6UzUF+h/U8218XfrauKNGmNKDYUhtmyBrHT+k6J0hQ4pN7fe6h+Z
w9RVHm24BGh1LxLHLms0IxvbrF277uX9Dxu1HfKfu5D2kimTY7xSZDNLr2dt+kNY
/+iWdIEEFpPT0PLSILt52wP6stF+3QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBIE
6w+WWC2gCfoJ06c9HMyGLMFepqZmz1n5IcQt1h9iy07Vkm1wkJiZsMhXpk73zXf
TPxuXEacTX3S0Ea070IMCFwkus05f61e0yFTynHCzBgZ3U0UkRVZA3WcpbNB6Dwy
h7ysVlqyT9WZd7E0Ym5j5oue2G2xdei+6etgn5UjyWm6liZGrc0F6WPTdmzqa6WG
ApEqanpkQd/HM+hUYex/ZS6zEhd4CCDLgYkIjlrFbFb3pJ10VLztIfSN5J40olpu
JVCfIq5u1NkpzL7ys/Ub8eYipbzI6P+yxXiUSuF0v9b98ymczMYjrSQXIf1e8In3
OP2Cc1CHoZ8XDQcvvKAh
-----END CERTIFICATE-----

```

Asia Pasifik (Osaka) — ap-northeast-3

DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxZzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxZzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0BAQ0BMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCySfYDk4mZr0LBA4GEAAKBgEbmveve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUHTRhxHhBZF0GvTFKxHoy9+f5H18wDQYJKoZIhvcNAQEL

```



```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJANuCGcCht0JhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA5MTQx
NTU3NDRaGA8yMTk1MDIxNzE1NTc0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWV2VydmljZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEAg66iNv6pJPMGM20W8HbVYJS1KcAg2vUGx8xeAbzZIQdpGfkabVcUHGB6m
Gy59VXDMD1rJckDDk6dxU0hmcX9z785TtVZURq1fua9QosdbTzX4kAgHGdp4xQEs
m06QZqg5qKjBP6xr3+PshfQ1rB8Bmwg0gXEm22CC7o77+7N7Mu2sWzWbiUR7vil4
9FjWS8XmMNwFT1Shp4l1TDTevDWW/uYmC30RThM9S4QPvTZ0rAS18hHVam8BCTxa
LHaVCH/Yy52rsz0hM/F1ghnSnK105ZKj+b+KIp3adBL80MCjgc/Pxi0+j3HQLdYE
32+FaXWU84D2iP2gDT28evnstzuYTQIDAQAABMA0GCSqGSIb3DQEBCwUAA4IBAQC1
mA4q+12pxy7By6g3nBk1s34PmWikNRJBw0qhF8ucGRv8aiNhRRye9lokXomwo8r
KHbbqvtK8510xUZp/Cx4sm4aTgcMvfJP29jGLc1DzeqADIVkWEJ4+xncxSYV1S9x
+78TvF/+8h9U2LnS164PXaKdxHy2IsHIVRN4GtoaP2Xhpa1S0M328Jykq/571nfN
1WRD1c/fQf1edgzRjhQ4whcAhv7WRRF+qTbfQJ/vDxy81ki0svU9XzUaZ0fZSfXX
wXxZamQb0NvFcxVHY/0PSiM8nQoUmkkBQuK1eDwRWvkoJKYKy13jvXK7HIWtM104
jmXe0aMy3thyK6g5sJVg
-----END CERTIFICATE-----
```

Asia Pasifik (Singapura) — ap-southeast-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0AQMDFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0AQBMIIBHwKBQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmve5f8LIE/Gf
```

```
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUSqP6ih+++5KF07NXngrWf26mhSUwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVjZjZlZm1jZXMgTEEx
MB4XDTE0MDQyOTE0MzAxNFoXDTI1MDQyODE0MzAxNFowXDELMAkGA1UEBhMVCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVjZjZlZm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdT
ZWV0dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUSqP6ih++
+5KF07NXngrWf26mhSUwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQAw13Bxw11U/JL58j//Fmk7qqtrZTqXmaz1qm2WlIpJpW750M0cP4ux1uPy
eM0RdVZ4jHSMv5gtLAv/PjExBfW9n6vNck+5GZG4Xec5DoapBZXHmfMo93sjxBFP
4x9rWn0GuwAV09ukjYPevq2Rerilrq5VvppHtbATVNY2qecXDA==
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAjVMGw5SHkcvMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWV0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEwMjkw
ODU3MTlaGA8yMTk1MDQwMzA4NTcxOVowXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVjZjZlZm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAlaSSLfB170gmikjLReHuNhVuvM20dCsVzptUyRbut+KmIEEc24wd/xVy
2RMIrydGedk4tUjkUy0yfET50AyT43jTzDPHZTkRSVkJYjBdcYbe9o/0Q4P7IVS3
X1vwrUu0qo9nSID0mxMn0oF118KAqnn10tQ0W+1NSTkasW7QVzcb+3okPEVhPA0q
Mn1Y3vkMQGI8zX4i0KbEcSVIzf6wuIffXMGHVC/JjwihJ2USQ8fq6oy686g54P4w
R0g415kLYcodjqThmGJPNUPAZ7M0c5Z4pymFuCHgNAZNVjhZDA8420jecqm62zcm
Tzh/pNMNeGCRYq2EQX0aQtY0Ij7b0QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
```

```
BgNVHQ4EFgQU6SSB+3qALorPMVNjToM1Bj3oJMswgY4GA1UdIwSBhjCBg4AU6SSB
+3qALorPMVNjToM1Bj3oJMuhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHEwdTZWF0dGx1MSAwHgYDVQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAJVMGw5SHkcvMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAF/0dWqkIEZK5rca8o0P0VS+to1JJE/FRZO
atH0eaQbWzyac6NEwjYeeV2kY63skJ+QPuYbSuIBLM8p/uTRIVYM4LZYImLGuvo0
IdtJ8mAzq8CZ3ipdMs1hRqF5GRp8l94w2QpX+PfhNw47iIOBiqSAUkIr3Y3BDaDn
EjeXF6qS4iPIvBaQQ0cvdddNh/pE33/ceghbkZNTYkrwMyBkQ1RTTVKXFN7pCRUV
+L9FuQ9y8mP0BYZa5e1sdkwebyU+eqVzsi198ntkhpjvRkaJ5+Drs8TjGaJW1Rw
5Wu0r8unKj7YxdL1bv7//RtVYVVi296ldoRUYv4SCvJF11z00dQ=
-----END CERTIFICATE-----
```

Asia Pasifik (Sydney) — ap-southeast-2

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQKIEExBXNoaW5ndG9uIFN0YXR1MRAwDgYDVQHEwdTZWF0dGx1MSAwHgYD
VQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExBXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQHEwdTZWF0dGx1MSAwHgYDVQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz11r7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
1Ra2v1ntMX3carVdDbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFXWYAdk4oiXI0C9PxcgjYYh71mwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0F0FtYXpvbiBXZWVlU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE1MjE0M1oXDTE1MDQyODE1MjE0M1owXDELMAkGA1UEBhMCVVMx
```

```

GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAE
BgNVBAoTF0FtYXpvbiBxZWlgaU2VydmVjZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZlwpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUFxWyAdk4
oiXI0C9PxcgjYYh71mwWegYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQByjeQe6lr7fiIhoGdjBXYzDfkX01GGvMIhRh57G1bbceQfaYdZd7PtC0j1
bpycKGA1UdIwUkpM0iV2Hi9d00YawkdhyJDstmDNKu6P9+b6Kak8He5z3NU1tUR2Y
uTwcZ7Ye8Nldx//ws3raErFTI7D6s9m630X8cAJ/f8bNgikwpw==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIIEjCCAvqgAwIBAgIJAL2b0gb+dq9rMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEwMjkw
OTAwNTdaGA8yMTk1MDQwMzA5MDA1N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAEBgNVBAoTF0Ft
YXpvbiBxZWlgaU2VydmVjZXMgTExDMiIIBiJANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAmRcyLWraysQS8yDC1b5Abs3TUaJabjqWu7d5gHik5Icd6dK18EYpQSeS
vz6pLhkg04xBbCRGlge8LS/OijcZ5HwdrxBiKbicR1YvIPaIyEQQvF5sX6UWkGYw
Ma5IRGj4YbRmJkBybw+AAV9Icb5LJNOMWpi340WM+2tMh+8L234v/JA6ogpdPuDr
sM6YFHMZ0NWo58MQ0FnEj2D7H58Ti//vFP10TaaPwaAIRF85zBiJtKcFJ6vPidqK
f2/SDuAvZmyHC8ZBHg1moX9bR5FsU3Qazfbw+c+JzAQWHj2AaQrGSCITxCM1S9sJ
151DeoZBjnx8cnRe+HCaC4YoRbiqIQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU/wHIo+r5U31VIsPoWoRVsNXGxowwgY4GA1UdIwSBhjCBg4AU/wHI
o+r5U31VIsPoWoRVsNXGxoyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIEExB
XYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IjAL2b0gb+dq9rMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAcCobLvj8Ix1QyORTz/9q7/VJL509/p4HAeve
92riHp6+Moi0/dSEYPeFTgdWB9W3YCnc34Ss9TJq2D7t/zLGG1bI4wYXU6VJjL0S
hCjWeIyBXUZ0ZKfCb0DSJeUElsTRSXSfUvRZ9EAwjLvHni3BaC9Ve34iP71ifr75
8Tpk6PEj0+JwiiJFH8E4GhcV5chB0/iooU6ioQqJrMwFYnwo1cVZJD5v6D0mu9bS
TMIJLJKv4QQQpPsNdjiB7G9bfbk6trP8fUVYLHLsV1Iy51Gx+tgwFEYkG1N8I00/
2LCawwaWm8FYAFd3IZ104RImNs/IMG7VmH1bf4swH0BHgCN1uYo=
-----END CERTIFICATE-----

```



```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL9KIB7Fgvg/MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
OTAwMjVhGA8yMTk1MDEyNzA5MDAyNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBhZG9zIL0gMlU+QmRSR0PH2Pfv9iejfLak9iwdm1WbWRrCEAj5VxPe0Q+I
Kezn0txzqQ5Wo5NLE9bA61sziUAFNVsTFUzphEwRohcekYyd3bBC4v/RuAjCXHVx
40z6AIksnA0GN2VABM1TeMNVPItKOCIErL111SqXX1gbtL1gxSW40JWdF3WPB68E
e+/1U3F70Er7XqmN0D0L6yh92QqZ8fHjG+af0L9Y2Hc4g+P1nk4w4iohQ0PABqzb
MPjK7B2Rze0f90Ec51GBQu13kxkWwQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU5DS5IFdU/QwYbikgtWvkU3fDwRgwY4GA1UdIwSBhjCBg4AU5DS5
IFdU/QwYbikgtWvkU3fDwRihYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExB
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAL9KIB7Fgvg/MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG/N7ua8IE9IMyno0n5T57erBvLT0Q79fIJN
Mf+mKRM7qRRsdg/eumFft0rL0Ko54pJ+Kim2cngCWNhkcctRHBV567AJNt4+ZDG5
hDgV0IXw01+eaLE4qzqWP/9Vr0+p3reuumGFZLVpvVpwXBBEBFUf2drUR14aWfI2
L/6VGINXYS7uP8v/2VBS7r6XZRnPBuY/R4hv5efYXnjwA9gq8+a3stC2ur8m5yS1
faKSWE4H320yAyaZWH4gpwUdbU1YgPHtm/ohRtiWPrN7KEG5Wq/REzMIjZCnx0fS
6KR6PNjlhxBsImQhmBvz6j5PLQx0xBZIpDoiK278e/1Wqm9LrBc=
-----END CERTIFICATE-----
```

Kanada (Pusat) — ca-central-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJhFw0z
ODAxMDUxMjU2MTJhMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkJ00AQBMIIbHwKBGQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz11r7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
-----END CERTIFICATE-----
```

```

hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUIrLgixJJB5C4G8z6pZ5rB0JU2aQwDQYJKoZIhvcNAQEL
BQAwXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE1MzU0M1oXDTE1MDQyODE1MzU0M1owXDELMAKGA1UEBhMVCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCCHvRjF/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBKTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdT
ZWf0dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUIrLgixJJ
B5C4G8z6pZ5rB0JU2aQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBHIQJmzyFAaSYs8SpiRijIDZW2RIo7qBkb/pI3rqK6y0WD1PuMr6yNI81D
IrkGfGftg4Z+2KETyU4x76HSf0s//vfH3QA57qFaAwddhKYy4BhteFQ1/Wex3xTLX
LiwI07kwJvJy3mS6UfQ4HcvZy219tY+0iy0Wrz/jVxwq7T0kCw==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJAJNKhJhaJ0uMMA0GCSqGSIb3DQEBwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDQoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDQoQHEwdTZWf0
dGx1MSAwHgYDQoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA3Mjkx
MTM3MTdaGA8yMTk2MDEwMjExMzU0M1owXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACeTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4OCQAQ8AMIIB
CgKCAQEAhDUh6j1ACSt057nSxAcwMaGr8Ez87VA2RW2HyY819XoHndnxmP50Cqld
+26AJt1t1qHpI1YdtnZ60rVgVhXcVtbvte01Z31dEzC3PMvmISBhHs6A3SWHA91n

```

```
InHbToLX/SWqBHL0X78HkPRaG2k0COHpRy+fG9gvz8HCiQaXCbWNFDHZev90ToNI
xhXBVzIa3AgUnGMa1CYZuh5AFVRCEeALG60kxMMC8IoAN7+HG+pMdqAhJxGUcM00
LBvmTGGeWhi04MUZwf0kwn9JjQZuyLg6B10D4Y6s0LB2P1MovmSJKGY4JcF8Qu3z
xxUb17Bh9pvzFR5gJN1pjM2n3gJEPwIDAQABMA0GCSqGSIB3DQEBCwUAA4IBAQAj
UNKM+gIIHNk0G0tzv6vZBT+o/vt+tIp81EoZwaPQh1121iw/I7ZvhMLAigx7eyvf
IxUt9/nf8pxWaeGzi98RbSmbap+uxYRynqe1p5rifTam0sguuPrhVp1120gRWLcT
rjg/K60UMXRsmg2w/cxV45pUBcyVb5h60p5uEVAVq+CVns13ExiQL6kk3guG4+Yq
LvP1p4DZfeC33a2Rfre2IHLsJH5D4SdWcYqBsftPf3FQThH010KoacGrXtsedsxs
9aRd70zuSEJ+mBxmzxSjSwM840oh78DjkdPQgv967p3d+8NiSLt3/n7MgnUy6WwB
KtDujDnB+ttEHwRRngX7
-----END CERTIFICATE-----
```

Kanada Barat (Calgary) — ca-barat-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAYPouptUMAKGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGMEFdhc2hpbmd0b24uU3RhdGUxED
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzriith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAoGAMITzTJUa6cBsIfdHN69zW/
aHjUB4r1ZfKb1FMhIp9EZtEf5n+06oXjUG2+dKRS1FQeEK333ehNZsPd6uqey6TYKtHpFb5XRLS8BpqB
+7gnbAd0CBZM5o4NWesSQ1GLnTdQcGZkYG/
QESkbadoCXQTifCujJE682hTDLIVt1d4ewwCQYHKoZiZjgEAwMvADAsAhRJc4gRS/HWTkCR2MESaQEe/
jOMNQIUNoTwLvUprMGpP1GiHe0veZi08=
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAzygAwIBAgIGAYPou9weMA0GCSqGSIB3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBYXNoaW5m
v4XBVH13ZCMgq1RHMqV8AWI5i06gFn2A9sN3AZXTMqwtZeIddebq3k6Wt7ieYvpXTg0qvgsjQIovRZwaBDBJy9x8C2hw
+w91MQjFhkJ7Jy/
PHCJ69EzebQIDAQABMA0GCSqGSIB3DQEBBQUAA4GBAGe9Snkz1A6rHBH6/5kDtYvtPYwhx2sXNxztbhkXErfk40Nw514
gvDVtWG7qyb6fAqgoisyAbk8K9LzxSim2S1nmT9vD84B/t/VvwQBylc
+ej8kRxMH7fquZLp7IXfmtBzyUqu6Dpbne+chG2
-----END CERTIFICATE-----
```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIIEejCCAvqgAwIBAgIJALyTn5IHrIZjMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0
dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMzEyMDcx
NTM3MDFaGA8yMjAzMDUxMzE1MzcwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBACTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCQAQ8AMIIB
CgKCAQEA1GP5os424BjMGPCK0Sg0c1P7lZUiB85du03M4hfjzS0szsBpmBGFDLz1
owYHtIx1q3+Vi1Lt5Q1x3id/ov1QyaBPFWXVek1HVXy9vieCcI3TdjGjT11W/8MM
m3X26QPcsnHM/Kk2wJ7s186MrqmdSsp3SCPpxv4vEG2Q9yR2bXY41hpc2rW1W8qU
D0JGX1uvmmAdFnto2011XWZ6xFen1h60DRugek/ufCbN+lJky0xLqPoavH0Ybjsb
UpsAsBs7phaon+X/5hIERfbp5Lfvnqq54pNG5Knu4KynfW9+kA/WS4cJ6FTTN5t+
y0P1HvcL+BL2RuDy6T2bB21xw5WqtQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQURTVu/Dd4zDnmS5G5CfVlnmUBN0swgY4GA1UdIwSBhjCBg4AURTVu
/Dd4zDnmS5G5CfVlnmUBN0uhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVoQIEExB
XNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYDVoQKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzAJALyTn5IHrIZjMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFt523A3Aug6/F8xxyITgA8gkU0btFh1XNSP
U4U20Q9n0tWI9WqnKNWH3KBxwY5EPitU6b3LM4xc9lDwPz7h2Pto+WhxP9LVKe6f
r8r7teTLCVZ7cfYZHzHg+f1ZjVpAgzE5BVfrr1j3QKpv0hYT3J1wMtI++Vorq5Nf
aPjzedehJLhmZVALwnfqfLrgv6/gmraP9Vmoa8U4D6A1jNiQGYaLwyoPoRm3bUs2
v1Mh9GkEQ1b9+1pFXcqqzJJTGRuiPCyPbECI79FAnx5JM/CkGJV8H10mjIW1qkK1
Y2qT7wzErrKLJyB53Pw15BdIM1onbDAQreZb0yZQLdoEl/tx7Uk=
-----END CERTIFICATE-----

```

Europa (Frankfurt) — eu-central-1

DSA

```

-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkiG9w0BAQ0DMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVoQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYD
VoQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVoQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVoQHEwdTZWF0dGx1MSAwHgYDVoQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkiG9w0BAQ0BMIIIBHwKBQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz11r7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U

```

```

hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUwXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFD5GsmkxRuecttwsCG763m3u63UwDQYJKoZIhvcNAQEL
BQAwXDELMAKGA1UEBhMCVVMxGTAXBgNVBAGTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBACsTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBZXWlGU2Vydm1jZXMgTEExD
MB4XDTE0MDQyOTE1NTUyOVowXDTI1MDQyODE1NTUyOVowXDELMAKGA1UEBhMCVVMx
GTAXBgNVBAGTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACsTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBZXWlGU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RwqWxCwB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcwWdUUIzvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcwWdUUIzvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWFOdGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUFD5Gsmkx
RuecttwsCG763m3u63UwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBh0WaX1BsW56Hqk588MmJxs0rvcKfDjF57RgEDgnGnQaJcStCVWD09UYO
JX2tdsPw+E7AjDqjsuxYaotLn3Mr3mK0sN0Xq9B1jBnWD4pARg89KZnZI8FN35HQ
0/LY0VHCknuPL123VmVRNs51qQA9hkPjvw21UzpdLxaUxt9Z/w==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAKD+v6LeR/WrMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWFO
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQw
OTA4MTlaGA8yMTk1MDExNzA5MDgX0VowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACsTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWlGU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAKa8FLhxs1cSJGK+Q+q/vTf8zVnDAPZ3U6oqpp0W/cupCtpwMAQcky8DY
Yb62GF7+C6usniaq/9W6xPn/3o//wti0cNt6MLsiUeHqN15H/4U/Q/fr+GA8pJ+L
npqZDG2tFi1WmvGhGgIbScrjR4V03TuKy+rZXYvMRk1RXZ9gPhk6evFnviwHsE
jV5AEjxLz3duD+u/SjPp1v1loxe2KuWnyC+EKInnka909s14ZAUh+qIYfZK85DAjm

```

```
GJP4W036E9wTJQF2hZJrzsiB1MGyC1WI9veRISd30izZZL6VVXLXUtHwVHnVASrS
zZDVpzj+3yD5hRXsvFigGhY0FCVFnwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUxC2l6pvJaRf1gu3MUdN6zTuP6YcwgY4GA1UdIwSBhjCBg4AUxC2l
6pvJaRf1gu3MUdN6zTuP6YehYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJAkD+v6LeR/WrMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAIK+DtbUPppJXFqQMv1f2Gky5/82ZwgbbfXa
HBeGSii55b3tsyC3ZW5Z1MJ7Dtnr3vUkiWbV1EUaZG0UIndUFtXUMABCb/coDndw
CAr53XTv7UwGVNe/AF0/6pQDdPxXn3xBhF0mTKPr0GdvYmjZUtQMSVb91bMwCFfs
w+SwDLnm5NF4yZchIcTs2fdpoyZp0HDXy0xgx01gWhKTnYbaZ0xkJvEvckcxVAwJ
obF8NyJ1a0/pWdjh1HafEXEN81yxyTTY0a0BGTuY0BD2cTYynauVKY4fqHUKr3v
Z6fboaHEd4RFamShM8uvSu6eEFD+qRmvq1codbpsS0huGNLzh0Q=
-----END CERTIFICATE-----
```

Europa (Irlandia) — eu-west-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEsBXIXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBXIXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQBMIIbHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GAAKBgEbmeve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDlwAwLAIUwXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUakDaQ1Zqy87Hy9ESXA1pFC116HkwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhZGUxEDA0
```

```
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTExD
MB4XDTE0MDQyOTE2MTgxMFoXDTI5MDQyODE2MTgxMFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQChvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJO+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnf0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQKIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUakDaQ1Zq
y87Hy9ESXA1pFC116HkwEgYDVR0TAQH/BAGwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQADIKn/MqaLGPuK5+prZZ50x4bBZLPtre02C7r0ppqU2kPM21VPyYYydkvP0
lgSmmsErGu/oL9JNztDe2oCA+kNy17ehcsf8cw0uP861czNFKCeU8b7FgBbL+sIm
qi33rAq6owWGi/5uEcFCR+JP7W+oSYYvir5r/yDmWzx+BvH5S/g==
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJA0rmqHuaUt0vMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQKIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0
dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEwMjkw
OTA2MTlaGA8yMTk1MDQwMzA5MDYxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAAQoCAQ8AMIIB
CgKCAQEAjE7nVu+aHLtZp9FYV25Qs1mvJ1JXD7J0iQ1Gs/RirW9a5ZECctc4ssnf
zQHq2JRVr0GRchvDrbm1HaP/avtFQR/Thvf1twu9AR0VT22dU0TvERdkNzveoFCy
hf52Rqf0DMrLXG8ZmQPPXPDFAv+sVMWCDftcChxRYZ6mP90+TpgYNT1krD5PdvJU
7HcXrkNHDYqbsg8A+Mu2hzl0QkvUET83Csg1ibeK54HP9w+FSD6F5W+6ZSHGJ88L
FI+qYKs7xsjJQYgXWfEt6bbckws1kZiAI0yMzYdPF6C1YzEec/UhIe/uJyUUNfpT
VIsI50ltBbcPF4c7Y20j0IwwI2Sg0QIDAQABo4HUMIHMAAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUF2DgPUZivKQR/Z18mB/MxIkjZDUwgY4GA1UdIwSBhjCBg4AUF2Dg
PUZivKQR/Z18mB/MxIkjZDWhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQKIEExB
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQKHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJA0rmqHuaUt0vMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAgM6+57W5brzJ3+T8/XsIdLTuiBSe5ALgSqI
qn05usUKAeQsa+kZIJPyEri5i8LEodh46DAF1R1XTMYgXXx10YggX88XPmPtok17
14hib/D9/lu4IaFIyLzYNSzsETyWkWoGve7ZFz60MTRTwY2u8YgJ5dec7gQgPSGj
avB0vTIgoW41G58sfw5b+wjXCsh0nR0on79RcQFFhGnvup0MZ+JbljyhZUYFzCli
31jPziKzqWa87xh2DbAyyvj2KZrZtTe2LQ48Z4G8wWytJzxEEzDREe4NoETf+Mu5G
4CqoaPR05KwkdNudGNwXewydb3+agdCgfTs+uAjeXKNdSpbhMYg=
-----END CERTIFICATE-----
```

Europa (London) — eu-west-2

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBGQCjKvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3Igb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUCgCV/DPxYNND/swDgEKGiC5I+EwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAClTB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzU2VydmljZXMgTEEx
MB4XDTE0MDQyOTE2MjJkxNFoXDTI1MDQyODE2MjJkxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAClTB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVzU2VydmljZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzIUCgCV/DPx
YNND/swDgEKGiC5I+EwwEgYDVR0TAAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQATPu/s0E2esNa4+XPEGK1EJSgqzyBSQLQc+VWo6FAJhGG9fp7D97jhHeLC
5vwfmtTAFnGBxadfa0T3ASKxn0ZhXtnRna460LtnNHm7ArCVgXKJo7uBn6ViXtFh
uEEw4y6p9YaLQna+VC8Xtgw6WKq2JXuKzuhuNKSFAgGw9vRcHg==
```

```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJANBx0E2b0CEPMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNjA4MTEw
NDU2NDJaGA8yMTk2MDExNTE0NTY0M1owXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWV2VydmljZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArYS3mJLGAmrh2DmiPLbqr4Z+xWXTzBWCj0wpsuHE9H6dWUuy12Bgnu+Z
d8QvW306Y1eec45M4F2RA3J4hWhTShzsM10JVRt+YulGeTf90CPr26QmIFfs5nD4
fgsJQEry2MBSGA9Fqx3Cw6qkWcr0PsCR+bH0U0XykdK10MnIbpBf0kTfciAupQEA
dEHnM2J1L2iI0NTLbgKxy5PXLH9weX20BFauNmHH9/J070pwL20SN5f8TxcM9+pj
Lbk8h1V4KdIwVQpdWkbDL9BCG1YjyadQJxSxz1J343NzrnDM0M4h4HtVaK0S7bQo
Bqt2ruopLRCYgcuFHck/1348iAmbRQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBg
wujwU10tpi3iBgmhjMClgZyMMn0aQIxMigoFNqXMUNx1Mq/e/Tx+SNa0EAu0n2FF
aiYjvY0/hX0x75ewzZvM7/zJWIdLdsgewpUq0BH4DXFhbSk2TxggSPb0WRqTBxq5
Ed7F7+7GRIeBbRzdLqmISDnfqey8ufW0ks51XcQNomDIRG5s9XZ5KHviDCar8FgL
HngBCdFI04CMagM+pwT09XN1Ivt+NzUj208ca3oP1IwEAd5KhIhPLcihBQA5/Lpi
h1s3170z1JQ1HZbDrH1pgp+8hSI0DwwDVb3IIH8kPR/J0Qn+hv012H0paUg2Ly0E
pt1RCZe+W7/dF4zsbqWk
-----END CERTIFICATE-----
```

Eropa (Milan) — eu-south-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqwCCQCME1HPdwG37jAJBgqhkiG9w0BAQ0MFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTA0MjkyMDM1MjJaFw00
NTA0MjkyMDM1MjJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbYwggErBgqhkiG9w0BAQ0BMIIIBHQBGAkoL4YfdMI/MrQ0oL
NPfeEk94eiCQA5xN0nU7+2eVQtEqjFbDADFENh1p3sh9Q90oheLFH8qpSfNDWn/0
ktCS909ApTY6Esx1ExjGSeQq/U+SC2JSuuTT4WFMKJ63a/czMtFkEPPnVIjJJJmT
HJSKSsVUgpdDIRvJXuyB0zdB+wIVALQ30LaVGd1PMNfS1nD/Yyn+32wnAoGAPBQ3
7XHg5NL0S4326eFRUT+4oInQFjJjP6dp3p0BEzpImNmZTtkCNNUKE4Go9hv5T41h
R0p0DvWv0CBupMAZVBP90bp1XPCyEIZtuDqVa7ukPOUpQNgQhLLAqkigTyXV0Smt
ECBj9tu5WNP/x3iTZTHJ+g0rhIqpgh012UwJpKADgYQAAGAV10EQPYQUg5/M3xf
```

```
6vE7jKTxxyFWEyjKfJK7PZCz0IGrE/swgACy4PYQW+AwcUweS1K/Hx20aZVUKzWo
wDUbeu65DcRdw2rSwCbBTU342sitFo/iGCV/Gjf+BaiAJtxniZze7J1lob8v0BeLv
uaMQmg0YeZ5e0f104GtqP1+1hcQwCQYHKoZIZjgEAwMwADAtAhQdoeWLrkm0K49+
AeBK+j6m2h9SKQIVAIBNhS2a8cQVABDCQXVXrc0t0m08
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJA0Z3GEIaDcugMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEsBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTEwMjQx
NTE5MDIaGA8yMTk5MMDMyOTE1MTkwOVowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudmVjZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQCjipGw3vsXRj4JoA16WQDyoPc/eh3QBARaApJEC4nPIGoUo1pAXcjFhWp1o20+
ivgfcsc4AU90pYdApha3spLey/bhHPri1JZHRNqScKP0hZsCNmKhfnZTIEQCFvsp
DRp4zr91/WS06/f1JFBYJ6JHhp0KwM81XQG591V6kkow7QIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAGLLrY3P+HH6C57dYgtJkuGZGT2+rMkk2n81/abzTJvsqRqGRrWv
XRKRX1KdM/dfiuYGokDGxiC0Mg6TYy6wvsR2qRhtXW10tZkiHwcQCn0ttz+8vpew
wx8JGMvowtuKB1iMsbwyRqZkFYLCvH+Opfb/Aayi20/ChQLdI6M2R5VU
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJA0/+DgYF78KwMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEsBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xOTA0Mjky
MDM1MjJaGA8yMTk4MTAwMjIwMzUyMTowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudmVjZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gCgKCAQEAv1ZLV+Z/P6INq+R1qLkzETBg7sFGKPiwHekbpuB61rRxKHhj8V9vaReM
lInv1Ur5LAPpMPYDsUJ4WoUbPYAqVqyMAo7ikJHCCM1cXgZJefgN6z9bpS+uA3YVh
V/0ipHh/X2hc2S9wvxKWiSHu6Aq9GVpqL035tJQD+NJuqFd+nXrtcw4yGtmvA6w1
5Bjn8WdsP3x0TKjrByYY1BhXpP/f1ohU9jE9dstsRXLa+XTgTPwCwdCS2oRTWPGR
c5Aeh47nnDsyQfP9gLxHeYeQItV/BD9kU/2Hn6mnRg/B9/TYH8qz1RTzLapXp4/5
iNwusrTNexG18BgvAPrfhjDpdgYuTwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB7
5ya11K/hKgvaRTvZwV8G1VZt0CGPtNv0i4AR/UN6Tmm51BzUB5nurB4z0R2MoY0
Uts9sLGvSFALJ4otoB77hyNpH3drttU1CVVwal/yK/RQLSon/IoUkaGEbqalu+mH
nYad5IG4tEbmeP456XXc058MKmnczNbPyw3FRzUZQtI/sf94qBwJ1Xo6XbzPKMy
xjL57LHIZCsd+XPifXay690FlsCIgLim11HgPkRIHE0XLSf3dsW9r+4CjoZqB/Z
jj/P4TLCxbYCLkvg1waMjgEWF40Img0fhx7yT2X92MiSrs3oncv/IqfdVTiN80Xq
-----END CERTIFICATE-----
```

```
jgnq1bf+EZEKvb6UCQV
-----END CERTIFICATE-----
```

Eropa (Paris) — eu-west-3

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgcqhkJ00AQDMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLcLnd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAKGBYqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUaC9fX57UDr6u1vBvsCsECKBZQyIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEEx
MB4XDTE0MDQy0TE2Mzcz0FoXDTE1MDQy0DE2Mzcz0FowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEyBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IUaC9fX57U
```

```

D1r6u1vBvsCsECKBZQyIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQCARv1bQEDaMEzYI0nPlu8GHcMXgmgA94HyrXhMMcaIlQwocGBs6VILGVhM
TXP2r3JFaPEpmXSQNQHvGA13c1KwAZbni8wtzv6qXb4L4muF34iQRHF0nYrEDoK7
mMPR8+oXKKuP0/mv/XKo6XAV5DDERdSYHX5kkA2R9wtvyZjPnQ==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJALWSfgHuT/ARMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNzA1MzEx
MTE4MTZaGA8yMTk2MTEwMzExMTg5N1owXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAY5V7KDqnEvF3D1SProFcgU/oL+QYD62b1U+Naq8aPuljJe127Sm9WnWA
EBd0SASk0aQ9fzjCPoG5SGgWkxYoZjsevHpmzjVv9+Ci+F57bSuMbjgUvbvRIFUB
bxQojVoXQPHgK5v4330DxkQ4sjRyUbf4YV1AFdfU7zabC698YgPV0ExGhXP1Tvco
8mlc631ubw2g52j0lzaozUkHPSbknTomhQIv06kUfX0e0TDMH4jLDG2ZIrUB1L4r
0WKG4KetduFrRZyDHF6ILZu+s6ywiMicUd+2U11DFC6oas+a8D11hm0/rpWU/ieV
jj4rWAFrsebnp+Nhgy96iiVUGS2LuQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDE
iYv6FQ6kXCg+sv1caQG9q59xUC5z8HvJZ1+SxzPKK4PKQdKvIIfE8GxVXq1ZG1
c15WKTFFDMapnzb9RV/DTaVzWx3cMYT77vm1H11XGjhx611CGcENH1egI310TILsa
+KfopuJEEQ9TDMAIkGjha+KieU/U5Ctv9fdej6d0GC60EuwKkTNzPWue6UMq8d4H
2xqJboWsE1t4nybEosvZfQJcZ8jyIYcYBnsG13vCLM+ixjuU5MVVQNMV/gBJzqJB
V+U0QiGiuT5cYgY/QihxdHt99zwGaE0ZBC7213NKr1NuLSrghDI2NLU8NsExq0Fy
OmY0v/xVmQUQ126jJXaM
-----END CERTIFICATE-----

```

Eropa (Spain) — eu-south-2

DSA

```

-----BEGIN CERTIFICATE-----
MIIC8DCCAq
+gAwIBAgIGAXjwLk46MAKGBYqGSM44BAMwXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVudm1jZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAY5V7KDqnEvF3D1SProFcgU/oL+QYD62b1U+Naq8aPuljJe127Sm9WnWA
EBd0SASk0aQ9fzjCPoG5SGgWkxYoZjsevHpmzjVv9+Ci+F57bSuMbjgUvbvRIFUB
bxQojVoXQPHgK5v4330DxkQ4sjRyUbf4YV1AFdfU7zabC698YgPV0ExGhXP1Tvco
8mlc631ubw2g52j0lzaozUkHPSbknTomhQIv06kUfX0e0TDMH4jLDG2ZIrUB1L4r
0WKG4KetduFrRZyDHF6ILZu+s6ywiMicUd+2U11DFC6oas+a8D11hm0/rpWU/ieV
jj4rWAFrsebnp+Nhgy96iiVUGS2LuQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDE
iYv6FQ6kXCg+sv1caQG9q59xUC5z8HvJZ1+SxzPKK4PKQdKvIIfE8GxVXq1ZG1
c15WKTFFDMapnzb9RV/DTaVzWx3cMYT77vm1H11XGjhx611CGcENH1egI310TILsa
+KfopuJEEQ9TDMAIkGjha+KieU/U5Ctv9fdej6d0GC60EuwKkTNzPWue6UMq8d4H
2xqJboWsE1t4nybEosvZfQJcZ8jyIYcYBnsG13vCLM+ixjuU5MVVQNMV/gBJzqJB
V+U0QiGiuT5cYgY/QihxdHt99zwGaE0ZBC7213NKr1NuLSrghDI2NLU8NsExq0Fy
OmY0v/xVmQUQ126jJXaM
-----END CERTIFICATE-----

```

```
+rzSaTaXE3B/R/4A2VuGqRYR7M1jPtwdmU6/3CPjCACcZmTIcOAKbFiDHqadQgBZXfzGpzw8Zo
+eYmmk5fXycgnj57PYH1dIWU6I7mCbAah5MZMcmHaTmIsomGrhcnWB8d8q0U7oZ0UWK4lbiAQs1MihoUwCQYHKoZIZjg
WmbaU7YM5GwCFCvIJ0es05hZ8PHC52dAR8WWC6oe
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAZYgAwIBAgIGAXjwLkiaMA0GCSqGSIB3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
VvR1+45Aey5zn3vPk6xBm5o9grSDL6D2iAuprQnfVXn8CIbSdbWFhA3fi5ippjKkh3s18VyCvCOUXKd0aNrYBrPRkrdH
+3m/
rxIUZ2IK1fDlC6sWAjddf6sBrV2w2a78H0H8EwuwiSggttURBjwJ7KPPJCqaqrQIDAQABMA0GCSqGSIB3DQEBBQUAA4GB
+FzqQDzun/
iMMzcFucMLM15BxEblrFX0z7IIu0eiGkndmrqUeDCykztLku45s7hxdNy41tTuVAaE5aNBdw5J8U1mRvsKvHLY2ThH6h
+hBgiphYp84DUBWVYeP8YqLEJSqscKscWC
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALWsm06DvSpQMA0GCSqGSIB3DQEBcUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFNlcnZpY2VzIEExMQzAgFw0yMjA3MTgx
MzU4NDNaGA8yMjAxMTIyMjEzNTg0M1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlgaU2Vydm1jZXMgTEExMIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAAuAAhuSpsHC00/fD2zN1BDpNLRndi9qbHsNeuz3WqN7Samj2aSrM2hS+i
hUxx0BspZj0tZC0sbpPZ+i74N0EQtFeqQoEGvKhB1nJiF4y5I81HDhs5qHvoIivm
7rbvik3zgm1PqS/DmDjVQaXPcD31Rd9ILwBmWEwJqHigyNV1xYtCzTQcrlBrvNZM
dnNgCDAdX/HBEFxx9012xeu0bSt0s+PJWZ1RTbYrNe7LIH6ntUqHxP/ziQ5trXEZ
uqy7aWk1L8uK4jmyNph0lbaqBa3Y6pYmU1nC27UE4i3fnPB0LSiAr+SrwVvX1g4z
i1o8kr+tbIF+JmcgYLBv08Jwp+EUqQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQUwvGzKJL9A5LReJ4Fxo5K6I20xcowgY4GA1UdIwSBhjCBg4AUwvGz
KJL9A5LReJ4Fxo5K6I20xcqhyKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFNlcnZpY2VzIEExMQ4IJALWsm06DvSpQMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAJAZd31jyoTGLawAD2+v/vQsaB9vZIx5EImi
G8YGkd61uFwEhAmtrwyE/i6FDSIphDrMHBkvw/D3BsqK+Ev/JOK/VYuaYDx/8fp
H4cwp9jC57CXzdIDREWNf6M9PsHFg2WA9XNNtC10ZL5WJiJwel8eDSg+sqJUxE01
MW+QChq/20F6niyaRK4bXrZq14as7h+F9u3A9xHE0VP7Zk9C2ehrBXzCMLSDt3GV
fEuMea2RxMhozWz34Hkdb6j18qoCfygubulovRNQjKw/cEmgPR16KfZPP5caILVt
9qkYPvePmbiVswZDee73cDymJYxLqILp0ZwyXvUH8StiH42FHZQ=
-----END CERTIFICATE-----
```

Europa (Stockholm) — eu-north-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqQCCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEEMQzCCAbcwggEsBgcqhkJ00AQBMIIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MbcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmve5f8LIE/Gf
MNMp9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKm11qx411HW
MXrs3Igb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDlwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAOqgAwIBAgIUN1c9U6U/xiVDFgJcYKZB4NkH1QEwDQYJKoZIhvcNAQEL
BQAwXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAClB1N1YXR0bGUxIDAEBgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEEx
MB4XDTE0MDQyOTE2MDYwM1oXDTE1MDQyODE2MDYwM1owXDELMAKGA1UEBhMVCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClB1N1YXR0bGUxIDAE
BgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTEExDMIGFMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvP12anJ0+eIB
UqPfQG09kZ1wpWpmy08bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13h0IcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQUJdbMCBXXtvCcWdwUuizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXXtvCcWdwUuizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTA1VTMRkwFwYDVQQIEiExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQ4IUN1c9U6U/
xiVDFgJcYKZB4NkH1QEwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBTIQdoFSDRHkppNPUbZ9WXR205v/9bpmHojMYZb3Hw46wsaRso7STiGGX/
tRqjIkPUIXsdhZ3+7S/RmhFznmZc8e0bjU4n5vi9CJtQSt+1u4E17+V2bF+D3h/7
wcfE013414Q8JaTDtEf/aF3F0uyBvr4MDMd7mFvAMmDmBPS1A==
```

```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJALc/uRxcg++EnMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWV6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xODA0MTAx
NDAwMTFaGA8yMTk3MDkxMzE0MDAxMVowXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBZXWV2VydmljZXMgTEExDMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAAzwCGJEJIxqtr2PD2a1mA6LhRzKhTBa1AZsg3eYfpETXIVlIrpjMfvVoN
qHvGshWlgrGTT6os/3gsaADheSaJKavxwX3X6tJA8fvEGqr3a1C1MffH9hBwbQqC
LbfUTAbkwis4GdTUw0WpJt1Cm3u9R/Vzi1CNwkj7iQ65AFAI8Enmsw3UGldEsop4
yChKB3KW3WI0FTh0+gD0YtjrqqYJxpG0YBpJp5vwdd3fZ4t1vidmDMs7liv4f9Bx
p0oSmUobU4GU1FhBchK1DukICVQdn0VzdMonYm7s+HtpFbVHR8yf6QoixBKGDsa1
mBf7+y0ixjCn0pnC0VLVooGo4mi17QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDG
40NZiixgk2sjJctwbyD5WKLTH6+mxYcDw+3y/F0fWz561YORhP2FNnPOmEkf0S1/
Jqk4svzJbCbQeMzRoyaya/46d7UioXMHRZam5IaGBh0dQbi97R4VsQjwQj0RmQsq
yDueDyuKTWwLK9KvI+ZA6e6bRkdNGf1K4N8GGKQ+fBhPwVELkbT9f160Jkezeen
S+F/gDADGJgmPXfjogICb4Kvshq0H5Lm/xZ1DULF2g/cYhyNY6E0I/eS5m1I7R8p
D/m6WoyZdpInxJfxW6160MkxQMRVsruLTNGtby3u1g6ScjmpFtvAMhYeJBsDzKG4
FEyxIdEjoe01jhTsck3R
-----END CERTIFICATE-----
```

Eropa (Zürich) — eu-central-2

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXiKJnMAKGBYqGSM44BAMwXDELMAkGA1UEBhMVCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLd1mVC1pJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmU1r7v60uqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAoGAYNjaCNg/
cfgQ011BUj5C1UulqwZ9Q+SfDzPZ9D2C0VbiRANiZoxrV8RdgmzzC5T7VcriVjwvta2Ch//
b+sZ86E5h0XWw1+BeEjD9cu3eDj12XB5sWEbNHNx49p5Tmtu5r2LDt1L8X/
Rpfalu2Z20JgjFJWGF7hRwx456n
+lowCQYHkoZIZjgEAWMvADAsAhRChsLcj4U5CVb2cp5M0RE1XbXmhAIUEGSnH+aiUQIWmPEFja+itWDufIk=
```

```
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
```

```
MIICMzCCAZygAwIBAgIGAXjSGFGiMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDBBXYXNoaW50
opKZAUusJx2hpgU3pUhh1p9ATh/VeVD582jTd9IY
+8t5MDa6Z3fGliByEiXz0LEHdi8MBacLREu1TwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAILlpoE3k9o7KdALAXsFJNi
+g3RMzdbiFM+7MA63Nv5fsf+0xgcjSNBE1vPCDKFvTJl4QQhToy0561105GvdS9RK
+H8xrP2mrqngApoKTApv93vHBixgFSn5KrczR00YSm30jkqbydU7DF1mkXXR7GYE+5jbHvQHYiT1J5sMu
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
```

```
MIIEEjCCAvqgAwIBAgIJALvT012pxTxNMA0GCSqGSIb3DQEBQwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0yMjA3MTgx
NTEyMDdaGA8yMjAxMTIyMjE1MTIwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACjTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWJgU2Vydm1jZXMgTEExMTIwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNV
CgKCAQEAYn+Lsnq1ykrfY1Zkk6aAAYNREnd9Iw8AUwCBkg0r2eBiBBepYxHwU85N
++moQ+j0EV2VaahBeTLShGZZS1HsyK8+cYT2QzpgHioamcYhrPXyIx1WiRQlaqSg
0FiE9bsqL3rCF5Vz+t0iTe5W/7ojf0Fls6++g7ZpobwJlpMbuJepqyeHMPyjv05A
age811Jewc4bxo2ntaW0HCqNksqfYB78j6X6kn3PFpX7FaYAwZA+Xx6C7UCY7rNi
UdQzfAo8htfJi4chz7frpUdQ9k13I0QrsLshBB5fFUjl09NiFipCGBwi+8ZMeSn1
5qwBI01BWPfG7WX60wyjhmh6JtE1wIDAQABo4HUMIHRMASGA1UdDwQEAwIHGDAd
BgNVHQ4EFgQU8HN4vvJrsZgPQeksMBgJb9xR1yYwgY4GA1UdIwSBhjCBg4AU8HN4
vvJrsZgPQeksMBgJb9xR1yahYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJALvT012pxTxNMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG1HYDtchPfbvdHx9HeQE8HgNugJUPdEqxun
t9U33p8VFrs+uLPtr0d9HDJEGvvs5h84EUie/oGJxRt7V1Vlid1PvHf6cRmpjgqY
YdggAVkZtY/PnFVmzf2bMV1SQPrqC17U0zaw2Kvnj4zgX0rZyCetgrZSUSxotyp
978Wy9ccXwVSeYG/YAr5rJpS6ZH7eRQvUY0IzwFNea0Pg0TEVpcjW1V6+MQEvsEx
W85q+s6AVr49eppEx8SLJs10C23yB+L+t32tAveQImRwtJmpzZ5cxh/sYgDVeOC0
85H1NK/7H9fAzT1cPu1oHSnB0xYzzHG0AmXmusMfwUk8fL1RQkE=
-----END CERTIFICATE-----
```

Israel (Tel Aviv) — il-central-1

DSA

```

-----BEGIN CERTIFICATE-----
MIIC7zCCAq+gAwIBAgIGAX0QPi
+9MAkGByqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACMB1NlYX
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNAfEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJfEnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZ16Ae1U1ZAFM0/7PSSoDgYQAAoGAbazCL5XXyPmcw3+oMYQUF5/9YogW6D0FZbYuyPgj0oUwWd16fj1zWca
pq+11ezuK2DF0zNTEyPEwwCQYHKOZIZjgEAWMvADAsAhRt1jKpXsvrS
+xTo2M9h2s2uLAhEQIU0Z2FcnTSrshF2EIdixZZwtNv66Q=
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIICmzCCAZygAwIBAgIGAX0QQGVLMA0GCSqGSIb3DQEjBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBXNoaW5n
+S8v0y5hpLoRe4Rk0rY0cM3bN07GdEMlin5mU0y1t8y3ct4YewvmkgT42kTyMM
+t1K4S0xsqjXxxS716uGyh7eWtkxrCihj8AbXN/6pa095h
+7TZy12n83keiNUz2M2KoqQVMwIDAQABMA0GCSqGSIb3DQEjBBQUAA4GBADwA6VVEIIZD2YL00F12po40xDLzIc9XvqFPS
FmU7H8s62/jD6c0R1A1cClIyZUe1yT1ZbPySCs43J+Thr8i8FSRxxzDBSZZi5foW
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJA0Vp1h2I9wW7MA0GCSqGSIb3DQEjBBQUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXNoaW5nNDg5uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbW6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0yMjA3MTUx
MjQ0MTJaGA8yMjAxMTIxOTExNDQxMjEwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBACMB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWJgU2Vydm1jZXMgTEExMjE1IjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEA13PkyWv161iV/SYf01UF076UpDfPm2SF/Rz/o33cm699X++EYPxTnoEc
vmWeS0I7eDxc40CUiToG0sEx0k1E0CX1Z1tK6qJ+zgWQLZ9SZEC9H0NsSA6LhrHu
Nq0dzeK3LjhdFcX46/4GqdiptpdTuM4m/h0Q5yx4JMQ/n1sdpv4M5VLRWwW9Lem
ufb79Id709SispxgRsz1KXIjp7N9S4BY7itSXz97uSyzTqEjWZ6mDUhTu3t21GKC
6f1ALGTTTrG2yghEhz53rkvLsvwzjPSS1T6LIfoMrRPzHaf+EdaKoasELE1SHh+ZH
9mI81HywE+HZ+W+5hBCvjYp90Y1fwIDAQABo4HUMIHRMASGA1UdDwQEAwIHgDAd

```

```
BgNVHQ4EFgQU58tN2J0+yEGq5JbIXxGi4vRVPyIwgY4GA1UdIwSBhjCBg4AU58tN
2J0+yEGq5JbIXxGi4vRVPyKhYKReMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEsxBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQ4IJA0Vp1h2I9wW7MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANBN0e1EqNy4+IU2yQzMJ+Wy5ZIOtTP6GSBR
7muVY1bDeAwtNTE0pwgrZV1C7/xq5Q0LC1y0Z70hHXEF8au7qStaAoUtxzvHTAZI
NC01woFU56UFw4N0vZII17iqEfoqRC4PpI30xqEJHFy0VL1vAzJoKB4QLLqDAYVA
LXCi0LoVT+y9tRYSxw5My00Bi6fxQIIAD12bE9xkunTN1Jkkwqo3LxNy/ryz4QWR
8K7jHUItifv4h/hxBKpHEquN8CkdvM9oeG17I8PFrSFEpGr1euDXy0euZzzYiDBV
m6GpTJgzpVsEuIX52dPcPemwQncoIfZyhWDW85MJUnby2WTEcFo=
-----END CERTIFICATE-----
```

Timur Tengah (Bahrain) — me-south-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQCVWlgSmP8RhTAJBgcqhkj00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEsBXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKEsxBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xOTAyMDUxMzA2MjFaFw00
NTAyMDUxMzA2MjFaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEsBXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEsxBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbggggEsBgcqhkj00AQBMIIbHwKBgQDcwojQfgWdV1Q1i00B
8n6cLZ38VE7ZmrjZ90QV//Gst6S1h7euhC23YppKXi1zovefSDwFU54zi3/oJ++q
PH1P1WGL8IZ34BUgRTtG4TVo1vp0smjkMvyRu5hIdKtZjV93Ccx15gVgyk+o1IEG
fZ2Kbw/Dd8JfoPS7KaSCmJKxXQIVAIzbIaDFRGa2qcMk2HWASyND17bAoGBANTz
IdhfMq+12I5iofY2oj3HI21Kj3LtZrWEg3W+/4rVhL31Tm0Nne1r19yGujrjQwy5
Zp9V4A/w9w2010Lx4K6hj34Eefy/aQnZwNdNhv/FQP7Az0fju+Y16L1300HQrL0z
Q+9cF7zEosekEnBQx3v6psNknKgD3Shgx+G0/LpCA4GFAAKBgQCVS7m77nuNA1Z8
wvUqcooxXMPkxJF154NxAu19KP9KN4svm003Zrb7t2F0tXRM8zU3TqMpryq1o5
mpMPsZDg6RXo9BF7Hn0DoZ6PJTamkFA6md+NyTJWJKvXC7iJ8fGDBJqTciUHuCKr
12AztQ8bFWsrTgTzPE3p6U5ckcgV1TAJBgcqhkj00AQDAy8AMCwCFB2NZGwM5ED1
86ayV3c1PEDukgQIAhQow38rQkN/VwHVeSW9DqEshXHjuQ==
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDPDCCAqWgAwIBAgIJAM16uIV/zqJFMA0GCSqGSIb3DQEBCwUAMHIXCzAJBgNV
BAYTA1VTMRMwEQYDVQQIDApYXNoaW5ndG9uMRAwDgYDVQQHEwAdTZWF0dGx1MSAw
HgYDVQQKDBBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzEaMBGGA1UEAwwRZWMyLmFt
YXpvbmF3cy5jb20wIBcNMTkwNDI2MTQzMjQ3WhgPMjE50DA5MjcxNDMyNDdaMHIX
```

```

CzAJBgNVBAYTA1VTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAAdTZWF0
dGx1MSAwHgYDVQQKDBdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzEaMBGGA1UEAwWR
ZWMyLmFtYXpvbmF3cy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALVN
CDTZEnIeoX1SEYqq6k1BV0ZlpY5y3Kno0reCAE589TwS4MX5+8Fzd6AmACmugeBP
Qk7Hm6b2+g/d4tWycyxLaQ1cq81DB1GmXehRkZRgGeRge1ePwD1TUA0I8P/QBT7S
gUePm/kANSFU+P7s7u1NNL+vynyi0wUUrw7/wIZTAgMBAAGjgdcwgdQwHQYDVR00
BBYEFILtMd+T4YgH1cgc+hVsV0V+480FMIGkBgNVHSMegZwwgZmAFILtMd+T4YgH
1cgc+hVsV0V+480FoXakdDBYMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGlu
Z3Rvb2EgMA4GA1UEBwwHU2VhdHRsZTEgMB4GA1UECgwXQW1hem9uIFd1YiBTZXJ2
aWN1cyBMTEMxGjAYBgNVBAMMEWVjMi5hbWF6b25hd3MuY29tggkAyXq4hX/OokUw
DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAA0BgQBhKNTBIFgWfd+Zhc/LhRUY
40jEiykmbEp6hlzQ79T0Tfbn5A4NYDI2icBP0+hmf6qSnIhwJF6typyd1yPK5Fqt
NTpxxcXmUKquX+pHmIkK1LKD08rNE84jqxrXrsfDi6by82fjVYf2pgjJW8R1FAw+
mL5WQRFexbfB5aXhcMo0AA==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJANZkF1QR2rKqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXHYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEEMQzAgFw0xOTAyMDUx
MzA2MjBaGA8yMTk4MDcxMTEzMDYyMFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAGT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWVgU2VydmljZXMgTEExIjIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAY4Vnit2eBpEjKgOKBmyupJzJAiT4fr74tuGJNwwa+Is2vH12jMzn9I11
UpvvEUyTIboIgISpf6Sj5LmV5rCv4jT4a1Wm0kjjfNbiI1kUi8SxZrPypcw24m6ke
BVuxQZrZDs+xDUYIZifTmdgD50u5YE+TLg+YmXKnVgxBU6WZjbuK2INohi71aPBw
2zWUR7Gr/ggIpf635JLU3KIBLNEmrkXCVSnDF1sK4eeCrB7+UNak+4BwgpuykSGG
Op9+2vsuNqFeU119daQeG9roHR+4rIWSPa0opmMxv5nctgyp0rE6zKXx2dNXQ1dd
VULv+WH7s6Vm4+yBeG8ctPYH5G0o+QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB5
ZcViiZdFdpcXESZP/KmZNDxB/kkt1IEIhsQ+MnN29jayE5oLmtGjHj5dtA3XNK1r
f6PVygvTKbtQLQqunRT83e8+7iCZMKI5ev7pITUQVvTUWI+Fc01JkYZxRF1VBuFA
WGZ0+98kxCS4n6tTwVt+nSuJr9BJRVC17apfHBgSS8c50Wna0VU/Cc9ka4eAfQR4
7pYSDU3wSRE01cs30q341XZ629IyFirSJ5TT0Ic0osNL7vmMQYj8H0n40BYqxKy8
ZJyvfXsIph0Na76PaBIs6ZlqA0f1LrjGzxBPiWRM/XrGmF8ze4KzoUqJEnK1306A
KHKgfiigQZ1+gv5FlyXH
-----END CERTIFICATE-----

```

Timur Tengah (UEA) — me-central-1

DSA

```

-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXhqnnMAkGByqGSM44BAMwXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s50f2EbdSP09EAMMeP4C2USZpRV1AI1H7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdirmVClpJ+f6AR7ECLCT7up1/63xhv401fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUA12BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v60uqC+VdMCz0HgmdRWVe0utRZT
+ZxBxCBgLRJFEnEj6EwoFh03zwyjMim4TwWeotUfI0o4K0uHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZL16Ae1U1ZAFM0/7PSSoDgYQAAoGAW+csuHsWp/7/
pv8CTKFwxsYudxuR6rbWahCkyIeAydXL9AWnphK6yp10DEMBF168Xq8Hp23s0WYf8mo0hqCom9+0+ovuUFdpvCie86bp
TOZU568Ty1ff3dDWbdRzeNQRHodRG+XEQSizMkAreeWt4kBa+PUwCQYHKOZIZjgEAWMvADAsAhQD3Z
+XGmzKmgALgGcVX/Qf1+Tn4QIUH1cgksBSVKbWj81tovBMJeKgdYo=
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjRrDjMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIDDBBYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
KyA6zyruJQrYy00a6wqLA7eeUzk3bMiTkLsTeDQfrkaZMfBAjGaa0ymRo1C3qzE4rIenmahvUp1u9ZmLwL1idWXMxR2R
+d2SeoK0KQWoc2U0FZMHYxDue7zkyk1CIRaBukTeY13/
RIr1c6X61zJ5BBtZX1HwayjQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBABTqTy3R6RXKPW45FA+cgo7YZEj/
Cnz5YaoUivRRdX2A83BHUBtvJE2+WX00FTEj4hRVjameE1nEno08Z7fUVloAFD1Do69fhkJeSvn51D1WRrPnoWGgEfr1
B+Wqm3kVEz/QNcz6npmA6
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM4h7b1CVhqqMA0GCSqGSIb3DQEBcQUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQZAgFw0yMjA0MTEy
MDE1MDNaGA8yMjAxMDkxNTEwMTUwMTowXDELMakGA1UEBhMCMVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVudm1jZXMgTEExDQYwMDE1MDkxNTEwMTUwMTUwMTUwMTUwMTUwMTUw
CgKCAQEApYbTWFm0hSoMpqPo72eqAmn1dXGZM+G8EoZXzWwT/+IHENB4q5N6k
tudYLre1bJxuzEw+iProSHjmb9bB9YscRTofjVhBl1t35Fc+i8BaMeH94SR/eE8Q0
m1l8gnLNW3d62lyuhzuyv1e5wV1RqzYw+X2zRH4/wRD0C0pzjKoHIgyPKsMgsw5
aTZhNMsGxZ9dbkf0iCGeQLDytwU/JTh/HqvSr3VfU0apTJJiyAxCtZWgp1/7wC
Rv0CSMRJobpUqxZgl/VsttwNkikSFz1wGkcYeSQvk+odbnYQckA8tdddoVI56eD4

```


Tiongkok (Beijing) – cn-north-1

DSA

```

-----BEGIN CERTIFICATE-----
MIIDNjCCA4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFAADBCMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFd1YiBTZXJ2aWNLcyBMTEMwIBcNMTUwNTEzMDk10TE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxZzA1BjBGNVBA1VTMRkwFwYDVQQIEExBXyXNo
aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6b24g
V2ViIFN1cnZpY2VzIEExMQzCCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/1
gsw9+QSnEJeYWnmivJW0Bdn9CyDpN7cpHVmeGgNjL2fvImWyWe2f2Kq/BL917N7C
P2ZT52/sH9orlck1n2z08xPi7MIItgPHQwu30xsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwH0/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
vtBj/SM4/IgQ3xJs1Fc190TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWfa0dy0+00aECAwEAATANBgkqhkiG9w0BAQsFAA0CAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7z1+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/70ynRyfQetJVGichaaxLNM31cr6kcx0owb+WQQ84cwrB3keykH4gRX
KHB2r1WSxta+2panSE01JX2q5jhcFP90rD0tZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZ1nIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+611MVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFhbQp1peGC19id0UqxPxWsasWxQX00azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----

```

RSA

```

-----BEGIN CERTIFICATE-----
MIIDCzCCANsGawIBAgIJALS0Mb0oU2svMA0GCSqGSIb3DQEBCwUAMFwxZzA1BjBGNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxZzA1BjBGNVBA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKEXdBbWF6
b24gV2ViIFN1cnZpY2VzIEExMQzCBnzANBgkqhkiG9w0BAQEFAA0BjQAwgYkCgYEA
uhhUN1qAZdcWwB/0SDVDGk30A99EFz0n/mJlmcIQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+e1KtJW/C0Sz/LYo60PUqd6iXF4h+upB9Hk00GuWHXsHBTsvgkgGA
1CGge14U0Cdq+23eANr8N8m28Uz1jjSnTlrYCHtzN4sCAwEA0B1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR00BBYEFBkZu3wT27NnYgrfH+xJz4HJanJoMIG0BgNVHSM
eGYYwgY0AFBkZu3wT27NnYgrfH+xJz4HJanJooWckXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFd1YiBTZXJ2aWNLcyBMTE0CCQ0jGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqzm117e8Hgb
oADS0ph+YUz5P/bUCm61wFj1xaTfwKcuTR3ytj7bFLow5Bm7Sa+TC1310Gb2taon

```

```
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUEk+MRiWu+0h5/1JGii3qw4YByx
SUDlRyNy1jJFstEZj0hs
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAi0gAwIBAgIJA0trM5XLDSjCMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQx
MDAxNDJaGA8yMTk1MDExNzEwMDE0MlowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhZGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWVzIGU2Vydm1jZXMgTEExMDE1IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvBz+WQNdPiM9S+aUUL0QEriTmNDUurjLWlr7Sfa0JScBzis5D5ju0jh1
+qJdkbuGktFX50TWtm8pWhInX+hI0oS3exC4BaANoa1A3o6quoG+Rsv72qQf8LLH
sgEi6+LM1CN9TwnRK0ToEabmDKorss4zF17VSsbQJwcBSf0cIwbdRRaW9Ab6uJHu
79L+mBR3Ea+G7vSDrVIA8goAPkae6jY9WGw9Kxs0rcvNdQoEkqRVtHo4bs9fMRHU
Etphj2gh40bX1FN92VtvzD6QBs3CcoFWgyWGvzg+dNG5VCbsiiuRdmii3kciZ3H
Nv1wCcZoEAqH72etVhsuvNRC/xAP8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA8
ezx5LRjzUU9EYWyhyYIEShF1P1qDhs7F4L46/51c4pL8FPoQm5CZuAF31DJhYi/b
fcV7i3n++/ymQbCLC6kAg8DUB7NrcR0l15ag8d/JXGzcTCn1DXLXx1905fPNa+jI
0q5quTmdmiSi0taeaKZmyUdhrB+a7ohWdSdlokEI0tbH1P+g5y113bI2leYE6Tm8
LKbyfK/532xJPq09abx4Ddn89ZEC6vvWVNDgTsxERg992Wi+/xoSw3XxkgAryIv1
zQ4dQ6irFmXwCWJqc6kHg/M5W+z60S/94+wGTXmp+19U6Rkq5jVMLh16XJXrXwHe
4KcgIS/aQGVgjM6wivVA
-----END CERTIFICATE-----
```

Tiongkok (Ningxia) – cn-northwest-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aW50cyBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDUwMTYwOTU5MTVaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNo
aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24g
V2ViIFN1cnZpY2VzIEExMQzAgFw0xNTA4MTQxMDAxNDJaGA8yMTk1MDExNzEwMDE0
MlowXDELMAKGA1UEBhMVCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhZGUxEDA0
BgNVBAClTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBxZWVzIGU2Vydm1jZXMgTE
ExMDE1IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAMWk9vyppSmDU3AxZ2Cy
2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/1gsw9+QSnEJeYWnmivJW0
Bdn9CyDpN7cpHVmeGgNjL2fvImWyWe2f2Kq/BL917N7C
P2ZT52/sH9orlck1n2z08xPi7MItgPHQwu30xsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwH0/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
-----END CERTIFICATE-----
```

```
vtBj/SM4/IgQ3xJs1Fc190TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWFa0dy0+00aECAwEAATANBgkqhkiG9w0BAQsFAA0CAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7z1+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAcoLrVu/70ynRyfQetJVGichaaxLNM3lcr6kcx0owb+WQQ84cwrB3keykH4gRX
KHB2r1WSxta+2panSE01JX2q5jhcFP90rD0tZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFhBQp1peGC19id0UqxPxWsasWxQX00azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJALS0Mb0oU2svMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWw6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBX
YXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWw6
b24gV2ViIFN1cnZpY2VzIEExMQzCBnzANBgkqhkiG9w0BAQEFAA0BjQAwgYkCgYEA
uhhUNlqAZdcwWB/OSDVGk30A99EFz0n/mJlmcIQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+e1KtJW/C0Sz/LYo60PUqd6iXF4h+upB9Hk00GuWHXsHBTsvgkgGA
1CGge14U0Cdq+23eANr8N8m28Uz1jjSnTlrYCHtzN4sCAwEAAa0B1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR00BBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIG0BgNVHSME
gYYwgY0AFBkZu3wT27NnYgrfH+xJz4HJaNJoWcKXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFd1YiBTZXJ2aWw1cyBMTE0CCQ0jGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqzm117e8Hgb
oADS0ph+YUz5P/bUCm61wFj1xaTfwKcuTR3ytj7bFLow5Bm7Sa+TC1310Gb2taon
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUEk+MRiWu+0h5/1JGii3qw4YByx
SUD1RyNy1jJFstEZj0hs
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJAPu4ssY3B1zcMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQQIEExBXyXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWw6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xNTEyMDMy
MTI5MzJaGA8yMTk1MDUwODIxMjkzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWlG2Vydm1jZXMgTEExDMIIb1jANBgkqhkiG9w0BAQEFAA0CAQ8AMIIB
CgKCAQEAs0iGi4A6+YTLzCdIyP8b8SCT2M/6PGKwzKJ5XbSBol3gsnSwiFYqPg9c
uJPNbiy9wSA9vlyfWMD90qvTfiNrT6vewP813QdJ3EENZ0x4ERcf/Wd22tV72kxD
```

```
yw1Q3I10MH4b0ItGQAxU50tXCjBZEEUZoo0kU8RoUQ0U2Pq14NTiUpzWacNutAn5
HHS7MDc4lUlsJqbN+5QW6fFrcNG/0Mrib3JbwdFUNhrQ5j+Yq5h78HarnUivnX/3
Ap+oPbentv1qd7wvPJU556LZuhfqI0TohiIT1Ah+yUdN5osoaMxTHKKtf/CsSJ1F
w3qXqFJQA0VWsqjFyHXFI32I/G0upwIDAQABMA0GCSqGSIB3DQEBCwUAA4IBAQCn
Um00QHvUsJSN6KATbghowLynHn3wZSQsuS8E0C0pcFJFxP2SV0NYkERbXu0n/Vhi
yq5F8v4/bRA2/xpedLWmvFs7QWlomuXhSnYFkd33Z5gnXPb9vRkLwiMSw4uXls35
qQraczUJ9EXDhrv7VmngIk9H3YsxYr1DGEqh/oz4Ze4UL0gnfkauanHikk+BUEsg
/jsTD+7e+niEzJPihHdsVFDlud5pakEzyxovHwNJ1GS2I//yxrJFIL91mehjqEk
RLPdNse7N6UvSnuXc0okwu6l6kfzigGkJBxkcq4gre3szZFdCQCuioj7Z4xtuTL8
YMqfiDtN5cbD8R8ojw9Y
-----END CERTIFICATE-----
```

AWS GovCloud (AS-Timur) — -1 us-gov-east

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgqhkJ00AQDMFwxCzAJBgNVBAYTA1VTMRkw
FwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQKKExdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTA1VTMRkwFwYDVQQIEExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQKKExdBbWF6b24gV2ViIFN1
cnZpY2VzIEExMQzCCAbcwggEsBgqhkJ00AQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5006kK/n1Lz1lr7D8ZwtQP8f0Epp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLc1nd/YozqNNmgIyZecN7Eg1K9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nvwHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buyCU1CiYQk40KNHCcHfNiZbd1x1E9rpUp7bnF
lRa2v1ntMX3carVdDbtPEWmdxSCYsYFDk4mZr0LBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQ0Gx5ho8WqD+aTebS+k2tn92BBPqeZqpWRa5P/+jrdKml1qx41lHW
MXrs3IgiB6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXB1k40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIULVyrqjjwZ461qe1PCiShB1KCCj4wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0
BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0F0tYXpvbiBxZWVudm1jZXMgTEExDjE1
MB4XDTE0MDUwNzE1MjUzN1oXDTE0MDUwNzE1MjUzN1owXDELMAkGA1UEBhMCVVMx
```

```

GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBR09K80a1
esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtXQrSCexBxvqRqQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHGDAdBgNVHQ4EFgQU1ZXneBYnPVYXkHV1Vjg7918V
gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPVYXkHV1Vjg7918VgE+hYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdT
ZWf0dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQ4IULVyrqjjw
Z461qe1PCiShB1KCCj4wEgYDVR0TAAQH/BAGwBgEB/wIBADANBgkqhkiG9w0BAQsF
AA0BgQBfAL/YZv0y3zmVbXjyxQCsD1oeDCJjFKIu3ameEckeIWJbST9LMto0zViZ
puIAf05x6GQiEqfBmk+YmXJfcTmJB4Ebaj4egFlslJPSHyC2xuydHlr3B04IN0H5
Z2oCM68u6GGbj0jZjg7GJonkReG9N72kDva/ukwZKgq8zErQVQ==
-----END CERTIFICATE-----

```

RSA-2048

```

-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIJALPB6hxFhay8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIEExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWf0
dGx1MSAwHgYDVQQKEXdBbWF6b24gV2ViIFN1cnZpY2VzIEExMQzAgFw0xODA0MTAx
MjMyNDlaGA8yMTk3MDkxMzEyMzI0VowXDELMAKGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAClTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlgaU2Vydm1jZXMgTEExDMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAv9xsI9237KYb/SPWmeCVzi7giKNron8hoRDwlwwMC9+uHPd53UxzKLB
pTgtJWAPkZVxEdl2Gdhwr3SULoKcKmkqE61tVFrVuPT33La1UufguT9k8ZDDu09C
hQNHUdSVEuVrK3bLjaSsM0S7Uxmnn71YT990IREowvnbNBsBlcabfQTBV04xfUG0
/m0XUiUFj0xDBqbNzkeIb1W7vK7ydSjTfMS1jga54UAVXibQt9EAI7B8k912iLa
mu9yEjyQy+ZQICTuAvPUEWe6va2CHVY9gYQLA31/zU0VBKZPTNExjaqK4j8bKs1/
7d0V1so39sIGBz21cUBec1o+yCS5SwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBt
h02W/Lm+Nk0qsXW6mqQFsAou0cASc/vtGNCyBfoFNX6aKXsVCHxq2aq2TUKWENS+
mKmYu1lZVhB0mLshy1lh3RRoL30hp3jCwXytkWQ7ElcGjDzNGc0FArzB8xFyQNdK
MNvXDi/ErzgrHGSpvcvmGHi0hMf3UzChMwBIr6udoDlMbSI07+8F+jUJkh4X111Kb
YeN5fsLZp7T/6YvbFSPpmbn1YoE2vKtuGKx0bRrhU3h4JHdp1Ze11pZ61h5iM0ec
SD11SximGIYCjfZpRqI3q50mbxCd7ckULz+UUPwLrf0ds4VrVVSj+x0ZdY19P1v2
9shw5ez6Cn7E3IfzqNH0
-----END CERTIFICATE-----

```



```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
```

```
MIID0zCCAi0gAwIBAgIJANCOF0Q6ohnuMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTA1VTMRkwFwYDVQKIExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0
dGx1MSAwHgYDVQQKExdBbWw6b24gV2ViIFN1cnZpY2VzIExMQzAgFw0xNTA5MTAx
OTQyNDdaGA8yMTk1MDIxMzE5NDI0N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDA0BgNVBAcTB1N1YXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBxZWlU2Vydm1jZXMgTEExMjI0MjI0MjI0MjI0MjI0MjI0MjI0MjI0MjI0
CgKCAQEAzIcGTzNqie3f1o1rrqcFzGfbymSM2QfbTzDIOG6xXXeFrCDAm0q0wUhi
3fRCuoeh1K0WAPu76B9os71+zgF22dIDEVkpqHCjBrGzDQZXXUw0zhm+PmBUI8Z1
qvbVD4ZYhjCujWzrsX6Z4yEK7PEFjtf4M4W8euw0RmiNwjy+knIFa+VxK6aQv94
1W98URFP2fD84xedHp6ozZ1r3+RZSIFZs0iyxYsgiwTbesRMI0Y7LnkKGCiHQ/XJ
0wSISWaCddbu59BZeADnyh14f+pWaSQpQQ1DpXvZAVBYvCH97J1oAxLfh8xcwgSQ
/se3wtn095VBt5b7qTVj0vy6vKZazwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA/
S8+a9csfASKdtQU0LsBynAbsBCH9Gykq2m8JS7YE4TGvqlpnWehz78rFTzQwmz4D
fwq8byPk16DjdF9utqZ0JUo/Fxelxom0h6oievtB1SkmZJNbgc2WYm1zi6ptViup
Y+4S2+vWZyg/X1PXD7wyRWuETmykk73uEyeWFBYKCHWs09sI+6204Vf8Jkuj/cie
1NSJX8fkerfvLrZSHBYhxLbL+actVEo00tiyZz8GnhgWx5faCY38D/k4Y/j5Vz99
71UX/+fWHT3+1TL8ZZK7f0QWh6NQP10wTP9KtWqf0UwMIbgFQPoxkP00TWRmdmPz
W0wT0bEf9ouTnjG90Z20
```

```
-----END CERTIFICATE-----
```

Peran identitas instans

Setiap instans 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.

Kredensial peran identitas instans dapat diakses dari Layanan Metadata Instans (IMDS) di `/identity-credentials/ec2/security-credentials/ec2-instance`. Kredensialnya terdiri dari AWS temporary access key pair dan session token. 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 EC2 — [EC2 Instance Connect](#) menggunakan peran identitas instans untuk memperbarui kunci host untuk instance Linux.
- Amazon GuardDuty — [Runtime Monitoring](#) menggunakan peran identitas instans untuk memungkinkan agen runtime mengirim telemetri keamanan ke titik akhir VPC. GuardDuty
- AWS Security Token Service (AWS STS) - Kredensial peran identitas instance dapat digunakan dengan tindakan. AWS STS [GetCallerIdentity](#)
- AWS Systems Manager— Saat menggunakan [Konfigurasi Manajemen Host Default](#), AWS Systems Manager gunakan 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-0123456789example
```

Untuk informasi selengkapnya tentang ARN, lihat [Amazon Resource Names \(ARN\)](#) di Panduan Pengguna IAM.

Jalankan perintah pada instans Amazon EC2 Anda saat peluncuran

Saat meluncurkan instans Amazon EC2, Anda dapat meneruskan data pengguna ke instans yang digunakan untuk melakukan tugas konfigurasi otomatis, atau menjalankan skrip setelah instance dimulai.

Jika Anda tertarik dengan skenario otomatisasi yang lebih kompleks, Anda dapat mempertimbangkan AWS CloudFormation atau AWS OpsWorks. Untuk informasi selengkapnya, lihat berikut ini:

- [Melakukan deployment aplikasi di Amazon EC2 dengan AWS CloudFormation](#) di Panduan Pengguna AWS CloudFormation .
- [AWS OpsWorks Panduan Pengguna](#).

Pada instance Linux, Anda dapat meneruskan dua jenis data pengguna ke Amazon EC2: skrip shell dan direktif 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 berenkode base64 (untuk panggilan API).

Pada instance Windows, agen peluncuran menangani skrip data pengguna Anda. Bagian berikut mencakup perbedaan dalam bagaimana data pengguna ditangani pada setiap sistem operasi.

Cara Amazon EC2 menangani data pengguna untuk instans Linux

Dalam contoh berikut, perintah dari [server Install a LAMP di Amazon Linux 2](#) dikonversi ke skrip shell dan satu set arahan cloud-init yang berjalan saat instance diluncurkan. Dalam setiap contoh, tugas-tugas berikut dijalankan oleh data pengguna:

- Paket perangkat lunak distribusi diperbarui.
- Server web yang diperlukan, php dan paket mariadb diinstal.
- Layanan httpd dimulai dan diaktifkan melalui systemctl.
- `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](#)
- [Data pengguna dan konsol](#)
- [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 dengan AMI Amazon Linux 2. Selain itu, instruksi ini dimaksudkan untuk digunakan dengan Amazon Linux 2, dan perintah serta arahan mungkin tidak berfungsi untuk distribusi Linux lainnya. Untuk informasi lebih lanjut tentang distribusi lain, seperti dukungan mereka untuk cloud-init, lihat dokumentasi spesifik mereka.

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 menggunakan data pengguna untuk menjalankan skrip secara otomatis dengan setiap restart instans Amazon EC2 Linux saya?](#) di pusat AWS pengetahuan.

Skrip shell data pengguna harus dimulai dengan `#!` karakter dan jalur ke penerjemah yang ingin Anda baca skrip (umumnya `/bin/bash`) Untuk pengenalan tentang 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 kredensi yang sesuai yang diperlukan oleh skrip data pengguna untuk mengeluarkan panggilan API. Untuk informasi selengkapnya, lihat [Menggunakan profil instans](#) dalam 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.

Data pengguna dan konsol

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 pengguna.

Tentukan data pengguna instans saat peluncuran

Ikuti prosedur untuk [meluncurkan instans](#). Bidang Data pengguna terletak di bagian [Detail lanjutan](#) wizard peluncuran instans. Masukkan skrip shell Anda di bidang Data pengguna, dan kemudian selesaikan prosedur peluncuran instans.

Pada contoh skrip di bawah ini, skrip membuat dan mengonfigurasi server web kita.

```
#!/bin/bash
yum update -y
amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
yum install -y httpd mariadb-server
systemctl start httpd
systemctl enable httpd
usermod -a -G apache ec2-user
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 {} \;
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

Berikan waktu yang cukup bagi instans untuk meluncurkan dan menjalankan perintah dalam skrip Anda, lalu periksa untuk melihat apakah skrip telah menyelesaikan tugas yang Anda maksudkan.

Misalnya, di peramban web, masukkan URL file pengujian PHP yang dibuat skrip. 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 [Menambahkan aturan ke grup keamanan](#).

(Opsional) Jika skrip Anda tidak menyelesaikan tugas yang Anda harapkan, atau jika Anda hanya ingin memverifikasi bahwa skrip Anda selesai tanpa kesalahan, [hubungkan ke instans](#), periksa file log output cloud-init (`/var/log/cloud-init-output.log`), dan cari pesan kesalahan di output.

Untuk informasi debugging tambahan, Anda bisa membuat arsip multipart Mime yang menyertakan bagian data cloud-init dengan petunjuk berikut:

```
output : { all : '| tee -a /var/log/cloud-init-output.log' }
```

Arahan ini mengirimkan perintah output dari skrip Anda ke `/var/log/cloud-init-output.log`. Untuk informasi selengkapnya tentang format data cloud-init dan membuat arsip multi-bagian Mime, lihat [Format cloud-init](#).

Lihat dan perbarui data pengguna instans

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

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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 Contoh.
3. Pilih instans dan pilih Status instans, Hentikan instans. Jika opsi ini dinonaktifkan, baik instans sudah dihentikan maupun perangkat root-nya 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 [Menggunakan cloud-init di Amazon Linux 2 di Panduan Pengguna 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 <http://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 menggunakan data pengguna untuk menjalankan skrip secara otomatis dengan setiap restart instans Amazon EC2 Linux saya?](#) di pusat AWS pengetahuan.

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 sebuah instans dengan data pengguna

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 mengonfigurasi server web di Amazon Linux 2. Baris `#cloud-config` di atas diperlukan untuk mengidentifikasi perintah sebagai arahan cloud-init.

```
#cloud-config
repo_update: true
repo_upgrade: all

packages:
- httpd
- mariadb-server

runcmd:
- [ sh, -c, "amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2" ]
- systemctl start httpd
- sudo 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 [Menambahkan aturan ke 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 [Ambil data pengguna instans dari instans Anda](#).

Di Windows, Anda dapat menggunakan AWS Tools for Windows PowerShell alih-alih menggunakan file 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`. Dengan `run-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](#). Dengan `modify-instance-attribute`, AWS CLI tidak melakukan pengkodean base64 dari data pengguna untuk Anda.

- Di komputer Linux, gunakan perintah base64 untuk mengkode data pengguna.

```
base64 my_script.txt >my_script_base64.txt
```

- Di komputer Windows, gunakan perintah certutil untuk mengkode 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](#). Dengan `describe-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":
    "IyEvYmLuL2Jhc2gKeXVtIHVwZGF0ZSAteQpzZXJ2aWNlIGh0dHBkIHh0YXJ0CmNoa2NvbWZpZyBodHRwZCBvbg=="
  },
  "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 diencode 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
--//--
```

Bagaimana Amazon EC2 menangani data pengguna untuk instans Windows

Pada instance Windows, agen peluncuran default untuk versi sistem operasi Anda menangani data pengguna sebagai berikut.

- [EC2Launch v2](#) menjalankan skrip data pengguna di Windows Server 2022
- [???](#) menjalankan skrip data pengguna pada Windows Server 2016 dan 2019
- [???](#) menjalankan skrip data pengguna pada versi Windows Server sebelum Windows Server 2016

[Untuk contoh perakitan UserData properti dalam AWS CloudFormation template, lihat Base64 Encoded Property dan Base64 Encoded UserData Property dengan dan. UserData AccessKey SecretKey](#)

Untuk contoh menjalankan perintah pada instance dalam grup Auto Scaling yang bekerja dengan kait siklus hidup, lihat [Tutorial: Mengonfigurasi data pengguna untuk mengambil status siklus hidup target melalui metadata instans di Panduan Pengguna Auto Scaling Amazon EC2](#).

Daftar Isi

- [Skrip data pengguna](#)
- [Eksekusi data pengguna](#)
- [Data pengguna dan konsol](#)
- [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 berjalan di jendela Command Prompt (perintah batch) atau menggunakan WindowsPowerShell.

Jika Anda menentukan skrip batch dan skrip Windows, PowerShell 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 kredensi

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 Windows PowerShell skrip](#)
- [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 yang terpisah dengan tugas `executeScript EC2launch v2` pada tahap `UserData`, tambahkan tanda berikut ke data pengguna Anda.

```
<detach>true</detach>
```

Note

Tanda lepas tidak didukung pada 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 Windows PowerShell skrip

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 tidak perlu menentukan kredensial ke cmdlet, karena aplikasi yang berjalan pada instance menggunakan kredensial peran untuk mengakses sumber daya (AWS misalnya, bucket Amazon S3).

Tentukan PowerShell skrip Windows menggunakan `<powershell>` tag. Pisahkan perintah menggunakan jeda baris. Tag `<powershell>` tidak peka huruf besar/kecil.

Sebagai 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 yang terpisah dengan tugas `executeScript` EC2Launch v2 pada tahap `UserData`, tambahkan tanda berikut ke data pengguna Anda.

```
<detach>true</detach>
```

Note

Tanda lepas tidak didukung pada 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 YAMAL 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 mengkode sendiri data pengguna. Jika tidak, kesalahan akan dicatat tentang tidak dapat menemukan tanda script atau powershell yang akan dijalankan. Berikut ini adalah contoh yang mengkodekan 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 lebih lanjut tentang pengodean basis64, lihat <https://www.ietf.org/rfc/rfc4648.txt>.

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 [Reboot atau mulai berikutnya](#).

Skrip data pengguna dijalankan dari akun administrator lokal ketika kata sandi acak dibuat. Jika tidak, skrip data pengguna dijalankan dari akun Sistem.

Peluncuran instans

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

ItuC:\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

Reboot atau mulai berikutnya

Saat Anda memperbarui data pengguna instans, skrip data pengguna tidak dijalankan secara otomatis saat Anda melakukan boot ulang atau memulai instans. Namun demikian, Anda dapat mengaktifkan eksekusi data pengguna sehingga skrip data pengguna dijalankan satu kali saat Anda melakukan boot ulang atau memulai instans, atau setiap kali Anda melakukan boot ulang atau memulai instans.

Jika Anda memilih opsi Matikan dengan Sysprep, skrip data pengguna akan dijalankan lain waktu saat instans dimulai atau dimulai ulang, meskipun Anda tidak mengaktifkan eksekusi data pengguna untuk boot ulang atau pemulaian berikutnya. Skrip data pengguna tidak akan dijalankan pada boot ulang atau permulaan berikutnya.

Untuk mengaktifkan eksekusi data pengguna dengan EC2Launch v2 (AMI Pratinjau)

- Untuk menjalankan tugas dalam data pengguna saat boot pertama, atur `frequency` ke `once`.
- Untuk menjalankan tugas dalam data pengguna pada setiap boot, atur `frequency` ke `always`.

Untuk mengaktifkan eksekusi data pengguna dengan EC2Launch (Windows Server 2016 atau setelahnya)

1. Hubungkan ke instans Windows Anda.
2. Buka jendela PowerShell perintah dan jalankan perintah berikut:

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

3. Putuskan sambungan dari instans Windows Anda. Untuk menjalankan skrip yang diperbarui saat instans dimulai lagi nanti, hentikan instans dan perbarui data pengguna.

Untuk mengaktifkan eksekusi data pengguna dengan EC2Config (Windows Server 2012 R2 dan sebelumnya)

1. Hubungkan ke instans Windows Anda.
2. Buka `C:\Program Files\Amazon\Ec2ConfigService\Ec2ConfigServiceSetting.exe`.
3. Untuk Data Pengguna, pilih Aktifkan UserData eksekusi untuk memulai layanan berikutnya.
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 konsol

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 instans saat peluncuran

Ikuti prosedur untuk [meluncurkan instans](#). Bidang Data pengguna terletak di bagian [Detail lanjutan](#) wizard peluncuran instans. Masukkan PowerShell skrip Anda di bidang Data pengguna, dan kemudian 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 mengosongkan kotak centang Data pengguna sudah diencode base64, konsol Amazon EC2 akan melakukan encode base64 untuk Anda.

User data - optional [Info](#)

Enter user data in the field.

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

User data has already been base64 encoded

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 Contoh.
3. Pilih instans dan pilih Tindakan, status instans, Hentikan instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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:

Edit user data Info

Instance ID

 [i-0655799f982552ec9](#)

Current user data

User data currently associated with this instance

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

 **Copy user data**

New user data

This user data will replace the current user data

Modify user data as text
Add your user data below

Modify user data by importing a file
Description of importing a file and what will happen to it

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>>true</persist>
```

Input is already base64-encoded

Cancel

Save

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.

Data pengguna dan Alat untuk Windows PowerShell

Anda dapat menggunakan Alat untuk 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 [Ambil data pengguna instans dari instans Anda](#). Untuk informasi tentang data pengguna dan AWS CLI, lihat [Data 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 Anda, gunakan [New-EC2Instance](#) perintah. Perintah ini tidak melakukan encode 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 contoh, gunakan [Get-EC2InstanceAttribute](#) perintah.

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute
userData).UserData
```

Berikut ini adalah output contoh. Perhatikan bahwa data pengguna diencode.

```
PHBvd2Vyc2h1bGw
+DQpSZW5hbWUtQ29tcHV0ZXIgLlU51d05hbWUgdXNlci1kYXRhLXRlc3QNCjwvcG93ZXJzaGVsbD4=
```

Gunakan perintah berikut untuk menyimpan data pengguna yang diencode 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 menggunakan peran IAM, lihat [Melampirkan IAM role ke instans](#)

Note

Skrip ini gagal pada versi Windows Server sebelum 2008.

```
<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. Untuk informasi selengkapnya, lihat [Bekerja dengan tanda instans dalam metadata instans](#).

Note

Skrip ini gagal pada versi Windows Server sebelum 2008.

```
<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>
```

Connect ke instans EC2

Bagian Panduan Pengguna Amazon EC2 ini memberikan informasi untuk membantu Anda terhubung ke instans Amazon EC2 setelah meluncurkannya. Ini juga menyediakan informasi untuk membantu Anda menghubungkan instans Anda ke AWS sumber daya lain.

Topik

- [Hubungkan ke instans Linux Anda](#)
- [Hubungkan ke instans Windows Anda](#)
- [Terhubung menggunakan Session Manager](#)
- [Connect ke instans Anda menggunakan EC2 Instance Connect Endpoint](#)
- [Hubungkan instans EC2 Anda ke sumber daya AWS](#)

Hubungkan ke instans Linux Anda

Ada banyak cara untuk terhubung ke instans Linux Anda. Beberapa bervariasi tergantung pada sistem operasi mesin lokal tempat Anda terhubung. Lainnya, seperti EC2 instans Connect atau AWS Systems Manager Session Manager, tidak berbeda. Di bagian ini, Anda dapat mempelajari cara terhubung ke instans Linux Anda dan mentransfer file antara komputer lokal dan instans Anda.

Sebelum Anda terhubung ke instans Linux, selesaikan prasyarat berikut.

- [Dapatkan informasi tentang instans Anda](#)

- [Temukan kunci pribadi dan atur izin](#)
- [\(Opsional\) Dapatkan sidik jari instans](#)

Kemudian, pilih salah satu opsi berikut untuk terhubung ke instans Linux Anda.

Opsi untuk terhubung berdasarkan sistem operasi lokal Anda

- [Hubungkan dari mesin lokal Linux atau macOS menggunakan SSH](#)
- [Hubungkan dari mesin lokal Windows](#)

Opsi untuk terhubung dari sistem operasi lokal

- [Terhubung menggunakan Session Manager](#)
- [Hubungkan ke instans Linux Anda dengan EC2 Instance Connect.](#)

 Note

Untuk tips pemecahan masalah koneksi instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Untuk memecahkan masalah boot, konfigurasi jaringan, dan masalah lainnya untuk instans yang dibangun di [Sistem AWS Nitro](#), Anda dapat menggunakan [Konsol Serial EC2 untuk instans Amazon EC2](#)

Dapatkan informasi tentang instans Anda

Untuk mempersiapkan koneksi ke instans, dapatkan informasi berikut dari konsol Amazon EC2 atau dengan menggunakan AWS CLI.

The screenshot shows the Amazon EC2 console interface. At the top, there's a notification 'Successfully started i-...' and a 'Launch Instances' button. Below that is a table of instances with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. The 'Instance ID' and 'Public IPv4 DNS' columns are circled in red. Below the table, the details for instance 'i-05e...' are shown, with tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The 'Details' tab is active, showing fields like Instance ID, IPv6 address, Public IPv4 address, Private IPv4 addresses, Instance state, Private IP DNS name, Instance type, VPC ID, Subnet ID, and Public IPv4 DNS. The 'Public IPv4 DNS' field is circled in red.

- 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 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 meluncurkan instans menggunakan [wizard peluncuran instans](#), Anda mungkin telah mengedit bidang Tetapkan otomatis IP publik pada Pengaturan jaringan dan mengubah nilainya menjadi Nonaktifkan. Jika Anda menonaktifkan opsi Tetapkan otomatis IP publik, instans tidak diberi alamat IP publik saat diluncurkan.

- (IPv6 saja) Dapatkan alamat IPv6 instans.

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 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](#).

- Dapatkan 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 pada instans Linux Anda](#).

- Dapatkan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda:

AMI digunakan untuk meluncurkan instans	Nama pengguna default
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos atau ec2-user
Debian	admin
Fedora	fedora atau ec2-user
RHEL	ec2-user atau root

AMI digunakan untuk meluncurkan instans	Nama pengguna default
SEGAR	ec2-user atau root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Linux Rocky	rocky
Lainnya	Memeriksa dengan penyedia AMI

Temukan kunci pribadi dan atur izin

Anda harus mengetahui lokasi file kunci privat Anda untuk terhubung ke instans Anda. Untuk koneksi SSH, Anda harus mengatur izin sehingga hanya Anda yang dapat membaca file.

Untuk informasi tentang cara kerja pasangan kunci saat menggunakan Amazon EC2, lihat [Pasangan kunci Amazon EC2 dan instans Amazon EC2](#).

- Temukan kunci privat

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

[Jika Anda kehilangan kunci privat untuk instans yang didukung EBS, 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 \[Siklus hidup instans\]\(#\).](#)

[Prosedur ini hanya didukung untuk instans dengan volume root EBS. Jika perangkat root adalah volume penyimpanan instans, Anda tidak dapat menggunakan prosedur ini untuk mendapatkan kembali akses ke instans Anda; Anda harus memiliki kunci privat untuk tersambung ke instans.](#)

Untuk menentukan tipe perangkat root instans Anda, buka konsol Amazon EC2, pilih Instans, pilih instans, pilih tab Penyimpanan, dan di bagian Detail perangkat root, periksa nilai Tipe 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 terhubung ke instans yang didukung EBS 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 instans 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, baik instans sudah dihentikan maupun perangkat root-nya adalah volume penyimpanan instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan instans, pastikan untuk mencadangkannya ke penyimpanan persisten.

Langkah 4: Luncurkan instans sementara

New console

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 AMI yang didukung Amazon EBS-Backed](#).

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.
-

Old console

Pilih Luncurkan instans, lalu gunakan wizard peluncuran untuk meluncurkan instans sementara dengan opsi berikut:

- Pada halaman Pilih AMI, 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 AMI yang didukung Amazon EBS-Backed](#).
 - Pada halaman Pilih Tipe Instans, tinggalkan tipe instans default yang dipilih oleh wizard untuk Anda.
 - Pada halaman Konfigurasi Detail Instans, tentukan Zona Ketersediaan yang sama dengan instans asli. Jika Anda meluncurkan instans dalam VPC, pilih sebuah subnet di Zona Ketersediaan ini.
 - Pada halaman Tambahkan Tanda, tambahkan tanda Name=Temporary ke instans untuk mengindikasikan bahwa ini adalah instans sementara.
 - Pada halaman Tinjau, pilih Luncurkan. Pilih pasangan kunci yang Anda buat di Langkah 1, lalu pilih Luncurkan Instans.
-

Langkah 5: Copot volume root dari instans asli dan lampirkan ke instans sementara

1. Di panel navigasi, pilih Volume dan pilih volume perangkat root untuk instans asli (Anda sudah 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 terlebih dahulu menghentikan instans sementara sebelum Anda dapat melampirkan volume.

Langkah 6: Tambahkan kunci publik baru ke **authorized_keys** pada volume asli yang dipasang ke instans 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 instans 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 output contoh ini, *222* adalah ID pengguna dan *500* adalah 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. Copot volume dari instans sementara (Anda melepasnya di langkah sebelumnya): Dari konsol Amazon EC2, pilih Volume di panel navigasi, pilih volume perangkat root untuk instans asli (Anda sudah mencatat ID volume pilih di langkah sebelumnya), 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 instans dari instans asli, tentukan nama perangkat yang Anda catat sebelumnya di [Langkah 2](#) untuk lampiran perangkat root asli (`/dev/sda1` atau `/dev/xvda`), lalu pilih Lampirkan volume.

Important

Jika Anda tidak menentukan nama perangkat yang sama dengan lampiran asli, Anda tidak dapat memulai instans asli. Amazon EC2 memperkirakan volume perangkat 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 instans sementara, dan pilih Status instans, Akhiri instans.

Jika Anda terhubung ke instans Anda menggunakan Putty dan perlu mengonversi file `.pem` ke `.ppk`, lihat [Konversi kunci privat Anda menggunakan PuTTYgen](#) di topik [Hubungkan ke instans Linux Anda dari Windows dengan PuTTY](#) di bagian ini.

- Setel izin kunci pribadi Anda sehingga hanya Anda yang dapat membacanya
- Hubungkan dari macOS atau Linux

(Instance Linux) Jika Anda berencana untuk menggunakan klien SSH di komputer macOS atau Linux untuk terhubung ke instance Linux Anda, gunakan perintah berikut untuk mengatur izin file kunci pribadi 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 klik kanan pada `.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 man-in-the-middle serangan, 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 menyerang. 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 [Menginstal AWS Command Line Interface](#) 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 instans Anda, lalu pilih Actions, Monitor dan troubleshoot, Dapatkan log sistem.

AWS CLI

Di komputer lokal Anda (bukan pada instance yang Anda sambungkan), gunakan perintah [get-console-output](#) ().AWS CLI Jika outputnya besar, [Anda dapat menyalurkannya ke file teks](#), yang mungkin lebih mudah dibaca. Perhatikan bahwa Anda harus menentukan Wilayah AWS kapan Anda menggunakan AWS CLI, baik secara eksplisit atau dengan menyetel Wilayah default. Untuk informasi tentang cara mengatur atau menentukan Wilayah, lihat [Dasar-dasar konfigurasi](#) di Panduan Pengguna AWS Command Line Interface .

```
aws ec2 get-console-output --instance-id instance_id --query Output --output
text > temp.txt
```

2. Dalam output konsol, temukan sidik jari instance (host), yang terletak di bawah BEGIN SSH HOST KEY FINGERPRINTS. Mungkin ada beberapa contoh sidik jari. Ketika Anda terhubung ke instans 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 SHA256:14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY no comment (ECDSA)
ec2: 256 SHA256:kpEa+rw/Uq3zxaYZN8KT501iBtJ0IdHG52dFi66EEfQ no comment (ED25519)
ec2: 2048 SHA256:L816pepcA7iqW/jBecQjVZC1UrKY+o2cHLI0iHerbVc no comment (RSA)
ec2: -----END SSH HOST KEY FINGERPRINTS-----
ec2: #####
```

Note

Anda akan mereferensikan sidik jari ini saat Anda terhubung ke instance.

Hubungkan ke instans Linux dari Linux menggunakan SSH.

Anda dapat menggunakan Secure Shell (SSH) untuk terhubung ke instans Linux Anda dari mesin lokal yang menjalankan sistem operasi Linux atau macOS, atau Anda dapat menggunakan alat koneksi independen platform, seperti EC2 Instance AWS Systems Manager Connect atau Session

Manager. Untuk informasi selengkapnya tentang alat independen platform, lihat [Hubungkan ke instans Linux Anda](#).

Halaman ini menjelaskan cara menghubungkan instans Anda dengan klien SSH. Untuk terhubung ke instans Linux Anda dari Windows, lihat [Hubungkan dari Windows](#).

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 menghubungkan ke instans Linux](#).

Daftar Isi

- [Prasyarat koneksi SSH](#)
- [Hubungkan ke instans Linux Anda menggunakan klien SSH](#)
- [Mentransfer file ke instans Linux menggunakan klien SCP](#)

Prasyarat koneksi SSH

Sebelum Anda terhubung ke instans Linux, selesaikan prasyarat berikut.

Memeriksa status instans Anda

Setelah Anda meluncurkan instans, perlu beberapa menit sampai instans tersebut siap, sehingga Anda dapat terhubung dengannya. Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Anda dapat melihat informasi ini di kolom Pemeriksaan status di halaman Instans.

Dapatkan nama DNS publik dan nama pengguna untuk ter-connect ke instans Anda

Untuk menemukan nama DNS publik atau alamat IP dari instans Anda dan nama pengguna yang harus Anda gunakan untuk terhubung ke instans Anda, lihat [Dapatkan informasi tentang instans Anda](#).

Temukan kunci pribadi dan atur izin

Untuk menemukan kunci privat yang diperlukan untuk terhubung ke instans Anda, dan untuk mengatur izin kunci, lihat [Temukan kunci pribadi dan atur izin](#).

Instal klien SSH di komputer lokal Anda sesuai kebutuhan

Komputer lokal Anda kemungkinan telah menginstal klien SSH secara default. Anda dapat memverifikasi ini dengan mengetik `ssh` di baris perintah. Jika komputer Anda tidak mengenali perintah tersebut, Anda dapat menginstal klien SSH.

- Versi terbaru Windows Server 2019 dan Windows 10 - OpenSSH disertakan sebagai komponen yang dapat diinstal. Untuk informasi selengkapnya, lihat [OpenSSH di Windows](#).
- Versi Windows sebelumnya - Unduh dan instal OpenSSH. Untuk informasi selengkapnya, lihat [Win32-OpenSSH](#).
- Linux dan macOS X - Unduh dan instal OpenSSH. Untuk informasi selengkapnya, lihat <https://www.openssh.com>.

Hubungkan ke instans Linux Anda menggunakan klien SSH

Gunakan prosedur berikut untuk terhubung ke instans Linux Anda menggunakan klien SSH. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Hubungkan ke instans Anda menggunakan SSH

1. Di jendela terminal, gunakan `ssh` untuk terhubung ke instans. Anda menentukan jalur dan nama file kunci privat (`.pem`), nama pengguna untuk instans Anda, dan nama DNS publik atau alamat IPv6 untuk instans Anda. Untuk informasi selengkapnya tentang cara menemukan kunci privat, nama pengguna untuk instans Anda, dan nama DNS atau alamat IPv6 untuk sebuah instans, lihat [Temukan kunci pribadi dan atur izin](#) dan [Dapatkan informasi tentang instans Anda](#). Untuk terhubung ke instans Anda, gunakan salah satu perintah berikut.

- (DNS Publik) Untuk terhubung menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk terhubung menggunakan alamat IPv6 instans Anda, masukkan perintah berikut.

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-IPv6-address
```

Anda akan melihat tanggapan seperti berikut:

```
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 14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

2. (Opsional) Pastikan sidik jari di 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 menyerang. man-in-the-middle Jika cocok, lanjutkan ke langkah berikutnya.
3. 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.
```

Mentransfer file ke instans Linux menggunakan klien SCP

Salah satu cara untuk mentransfer file antara komputer lokal dan instans Linux Anda adalah dengan menggunakan protokol salinan aman (SCP). Bagian ini menjelaskan cara mentransfer file dengan SCP. Prosedurnya mirip dengan prosedur untuk menghubungkan ke sebuah instans dengan SSH.

Prasyarat

- Verifikasi prasyarat umum untuk mentransfer file ke instans Anda.

Sebelum mentransfer file antara mesin lokal dan instans Anda, lakukan tindakan berikut untuk memastikan bahwa Anda memiliki semua informasi yang dibutuhkan.

- [Dapatkan informasi tentang instans Anda](#)
- [Temukan kunci pribadi dan atur izin](#)
- [\(Opsional\) Dapatkan sidik jari instans](#)
- 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. Misalnya, jika nama file kunci privatnya adalah, `key-pair-name.pem` maka file yang akan ditransfer adalah, `my-file.txt` nama pengguna adalah, `ec2-user` dan nama DNS publik dari instans tersebut adalah, `instance-public-dns-name` dan alamat IPv6 dari instans tersebut adalah `instance-IPv6-address`.

- (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@[instance-IPv6-address]: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@[instance-IPv6-address]:path/my-file.txt path/my-file2.txt
```

Hubungkan ke instans Linux Anda dari Windows

Anda dapat menggunakan metode berikut untuk terhubung ke instans Linux Anda dari mesin lokal yang memiliki sistem operasi Windows.

Hubungkan ke instans Linux Anda dari Windows menggunakan OpenSSH

Prosedur berikut menunjukkan cara menghubungkan ke instans Linux Anda dari Windows menggunakan OpenSSH, alat konektivitas open source untuk login jarak jauh dengan protokol SSH. OpenSSH didukung pada Windows Server 2019 dan sistem operasi setelahnya.

Daftar Isi

- [Prasyarat](#)
- [Instal OpenSSH untuk Windows menggunakan PowerShell](#)
- [Hubungkan ke instans Linux Anda dari Windows menggunakan OpenSSH](#)
- [Copot pemasangan OpenSSH dari Windows menggunakan PowerShell](#)

Prasyarat

Sebelum Anda terhubung ke instans Linux menggunakan OpenSSH, lengkapi prasyarat berikut.

Verifikasi bahwa instans sudah siap

Setelah Anda meluncurkan instans, perlu beberapa menit sampai instans tersebut siap, sehingga Anda dapat terhubung dengannya. Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Anda dapat melihat informasi ini di kolom Pemeriksaan status di halaman Instans.

Verifikasi prasyarat umum untuk menghubungkan ke instans Anda

Untuk menemukan nama DNS publik atau alamat IP dari instans Anda dan nama pengguna yang harus digunakan untuk terhubung ke instans Anda, lihat [Dapatkan informasi tentang instans Anda](#).

Verifikasi versi Windows Anda

Untuk terhubung ke instans Linux Anda dari Windows menggunakan OpenSSH, versi Windows harus Windows Server 2019 dan seterusnya.

Verifikasi PowerShell prasyarat

Untuk menginstal OpenSSH di OS Windows Anda PowerShell menggunakan, Anda harus PowerShell menjalankan versi 5.1 atau yang lebih baru, dan akun Anda harus menjadi anggota grup Administrator bawaan. Jalankan `$PSVersionTable.PSVersion` 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`.

Instal OpenSSH untuk Windows menggunakan PowerShell

Untuk menginstal OpenSSH untuk Windows PowerShell menggunakan, jalankan perintah berikut:

PowerShell

```
Add-WindowsCapability -Online -Name OpenSSH.Client~0.0.1.0
```

Keluaran yang diharapkan

```
Path      :  
Online    : True  
RestartNeeded : False
```

Hubungkan ke instans Linux Anda dari Windows menggunakan OpenSSH

Setelah Anda menginstal OpenSSH, gunakan prosedur berikut untuk terhubung ke instans Linux Anda dari Windows menggunakan OpenSSH. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Untuk terhubung ke instans Anda menggunakan OpenSSH

1. Dalam PowerShell atau Command Prompt, gunakan ssh perintah untuk terhubung ke instance. Anda menentukan jalur dan nama file kunci privat (.pem), nama pengguna untuk instans Anda, dan nama DNS atau alamat IPv6 publik untuk instans Anda. Untuk informasi selengkapnya tentang cara menemukan kunci privat, nama pengguna untuk instans Anda, dan nama DNS atau alamat IPv6 untuk sebuah instans, lihat [Temukan kunci pribadi dan atur izin](#) dan [Dapatkan informasi tentang instans Anda](#). Untuk terhubung ke instans Anda, gunakan salah satu perintah berikut.

- (DNS Publik) Untuk terhubung menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk terhubung menggunakan alamat IPv6 instans Anda, masukkan perintah berikut.

```
ssh -i /path/key-pair-name.pem instance-user-name@instance-IPv6-address
```

Anda akan melihat tanggapan seperti berikut:

```
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 14UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.  
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

2. (Opsional) Pastikan sidik jari di 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 menyerang. man-in-the-middle Jika cocok, lanjutkan ke langkah berikutnya.

3. 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.
```

Copot pemasangan OpenSSH dari Windows menggunakan PowerShell

Untuk menghapus OpenSSH dari Windows PowerShell menggunakan, jalankan perintah berikut:
PowerShell

```
Remove-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

Keluaran yang diharapkan

```
Path          :
Online        : True
RestartNeeded : True
```

Hubungkan ke instans Linux Anda dari Windows dengan PuTTY

Jika Anda menjalankan Windows Server 2019 atau setelahnya, sebaiknya gunakan OpenSSH, yang merupakan alat konektivitas open source untuk login jarak jauh dengan protokol SSH. Untuk langkah-langkah menghubungkan ke instans Linux dari Windows menggunakan OpenSSH, lihat [Hubungkan ke instans Linux Anda dari Windows menggunakan OpenSSH](#).

Petunjuk berikut menjelaskan cara ter-connect ke instans Anda menggunakan PuTTY, klien SSH gratis untuk Windows. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Daftar Isi

- [Prasyarat](#)
 - [Konversi kunci privat Anda menggunakan PuTTYgen](#)
- [Hubungkan ke instans Linux Anda](#)
- [Mentransfer file ke instans Linux Anda menggunakan klien PuTTY Secure Copy](#)
- [Mentransfer file ke instans Linux Anda menggunakan WinSCP](#)

Prasyarat

Sebelum Anda terhubung ke instans Linux menggunakan PuTTY, lengkapi prasyarat berikut.

Verifikasi bahwa instans sudah siap

Setelah Anda meluncurkan instans, perlu beberapa menit sampai instans tersebut siap, sehingga Anda dapat terhubung dengannya. Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Anda dapat melihat informasi ini di kolom Pemeriksaan status di halaman Instans.

Verifikasi prasyarat umum untuk menghubungkan ke instans Anda

Untuk menemukan nama DNS publik atau alamat IP dari instans Anda dan nama pengguna yang harus digunakan untuk terhubung ke instans Anda, lihat [Dapatkan informasi tentang instans Anda](#).

Install PuTTY di komputer lokal Anda

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.

Konversi kunci .pem privat Anda ke .ppk menggunakan PuTTYgen

Untuk key pair yang Anda tentukan saat meluncurkan instans, jika Anda memilih untuk membuat kunci privat dalam format.pem, Anda harus mengonversinya menjadi file.ppk untuk digunakan dengan PuTTY. Temukan file.pem privat, lalu ikuti langkah-langkah di bagian berikutnya.

Konversi kunci privat 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. Anda harus mengonversi kunci privat Anda (file .pem) menjadi format ini (file .ppk) sebagai berikut untuk terhubung ke instans Anda menggunakan PuTTY.

Untuk mengonversi kunci .pem privat Anda menjadi .ppk

1. Dari menu Mulai, pilih Semua Program, PuTTY, PuTTYgen.
2. Di bawah tipe kunci yang akan dibuat, pilih RSA. Jika versi PuTTY Anda tidak menyertakan opsi ini, pilih SSH-2 RSA.

Parameters

Type of key to generate:

RSA DSA ECDSA EdDSA SSH-1 (RSA)

Number of bits in a generated key:

- Pilih Muat. Secara default, PuTTYgen hanya menampilkan file dengan ekstensi .ppk. Untuk menemukan file .pem, pilih opsi untuk menampilkan file dari semua tipe.

File name: ▼

PuTTY Private Key Files (*.ppk) ▼

PuTTY Private Key Files (*.ppk)

All Files (*.*)

- 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.
- 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.

- 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 [Konversi kunci privat Anda menggunakan PuTTYgen](#) di bagian sebelumnya. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

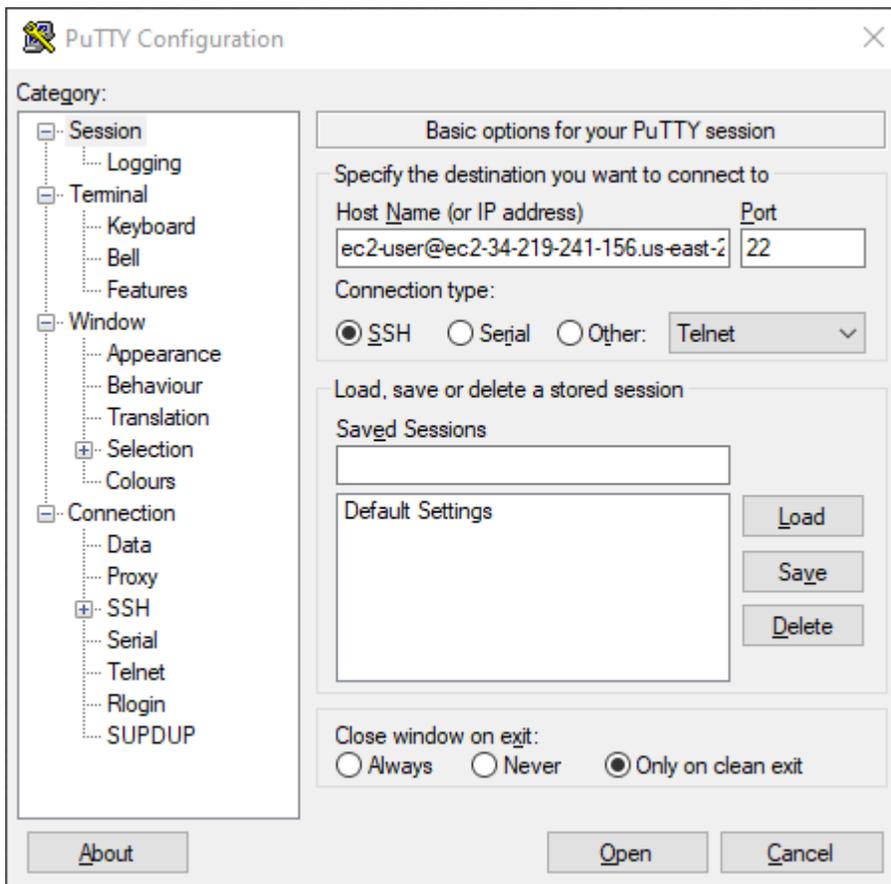
Versi PuTTY yang diuji terakhir: .78

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 terhubung 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@instance-IPv6-address*.

Untuk informasi tentang cara mendapatkan nama pengguna untuk instans Anda, dan nama DNS publik atau alamat IPv6 instans Anda, lihat [Dapatkan informasi tentang instans Anda](#).

- 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 mengklik dua kali pada file atau memilih Buka di kotak dialog Pilih file kunci privat.
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 menghubungkan ke instans Linux](#).

Mentransfer file ke instans Linux Anda menggunakan klien PuTTY Secure Copy

PuTTY Secure Copy Client (PSCP) adalah alat baris perintah yang dapat Anda gunakan untuk mentransfer file antara komputer Windows dan instans Linux Anda. Jika Anda lebih suka antarmuka pengguna grafis (GUI), maka Anda dapat menggunakan alat GUI sumber terbuka bernama WinSCP. Untuk informasi selengkapnya, lihat [Mentransfer file ke instans Linux Anda menggunakan WinSCP](#).

Untuk menggunakan PSCP, Anda memerlukan kunci privar yang Anda buat di [Konversi kunci privat Anda menggunakan PuTTYgen](#). Anda juga memerlukan nama DNS publik dari instans Linux Anda, atau alamat IPv6 jika instans Anda memilikinya.

Contoh berikut mentransfer file `Sample_file.txt` dari drive `C:\` pada komputer Windows ke direktori beranda `instance-user-name` pada instans Amazon Linux. Untuk mentransfer file, gunakan salah satu perintah berikut.

- (DNS Publik) Untuk mentransfer file menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
pscp -i C:\path\my-key-pair.ppk C:\path\Sample_file.txt instance-user-name@instance-public-dns-name:/home/instance-user-name/Sample_file.txt
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk mentransfer file menggunakan alamat IPv6 instans Anda, masukkan perintah berikut. Alamat IPv6 harus diapit tanda kurung siku ([]).

```
pscp -i C:\path\my-key-pair.ppk C:\path\Sample_file.txt instance-user-name@[instance-IPv6-address]:/home/instance-user-name/Sample_file.txt
```

Mentransfer file ke instans Linux Anda menggunakan WinSCP

WinSCP adalah pengelola file berbasis GUI untuk Windows yang dapat Anda gunakan untuk mengunggah dan mentransfer file ke komputer jarak jauh menggunakan protokol SFTP, SCP, FTP, dan FTPS. Dengan WinSCP, Anda dapat menarik dan melepas file dari komputer Windows ke instans Linux Anda atau menyinkronkan seluruh struktur direktori antara kedua sistem.

Persyaratan

- Anda harus memiliki kunci privat yang Anda hasilkan di [Konversi kunci privat Anda menggunakan PuTTYgen](#).
- Anda harus memiliki nama DNS publik dari instans Linux Anda.
- Instans Linux Anda harus menginstal scp. Untuk beberapa sistem operasi, Anda menginstal paket openssh-clients. Untuk yang lain, seperti AMI dengan pengoptimalan Amazon ECS, instal paket scp. Periksa dokumentasi untuk distribusi Linux Anda.

Untuk terhubung ke instans Anda menggunakan WinSCP

1. Unduh dan instal WinSCP dari <http://winscp.net/eng/download.php>. Bagi sebagian besar pengguna, opsi instalasi default tidak masalah.
2. Mulai WinSCP.
3. Pada layar Login WinSCP, untuk Nama host, masukkan salah satu dari berikut ini:
 - (DNS atau alamat IPv4 publik) Untuk masuk menggunakan nama DNS publik atau alamat IPv4 publik instans Anda, masukkan nama DNS publik atau alamat IPv4 publik untuk instans Anda.
 - (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk masuk menggunakan alamat IPv6 instans Anda, masukkan alamat IPv6 untuk instans Anda.
4. Untuk Nama pengguna, masukkan nama pengguna default untuk AMI Anda.

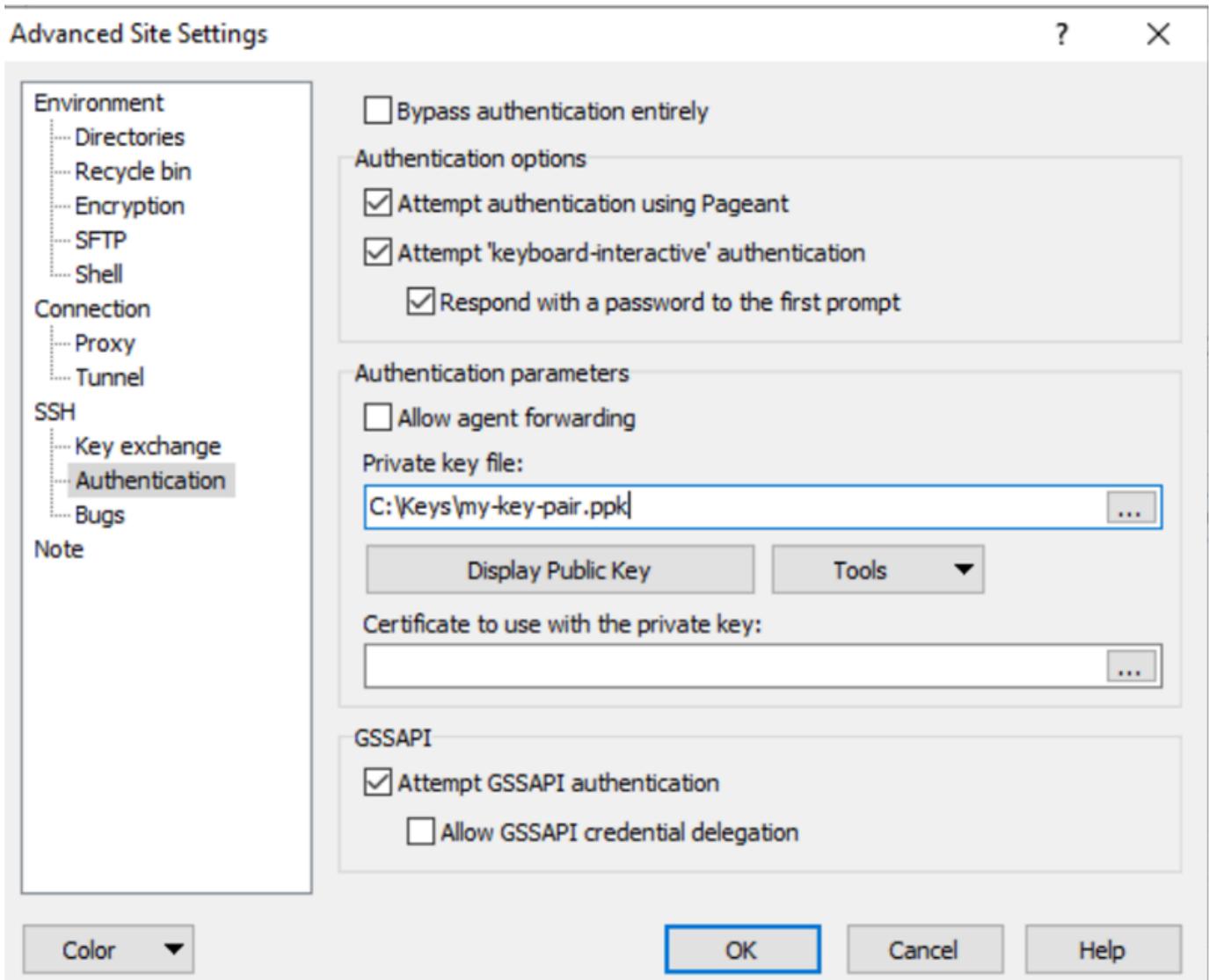
- Untuk AL2023, Amazon Linux 2, atau AMI Amazon Linux, nama penggunanya adalah `ec2-user`.
- Untuk AMI CentOS, nama penggunanya adalah `centos` atau `ec2-user`.
- Untuk AMI Debian, nama penggunanya adalah `admin`.
- Untuk AMI Fedora, nama penggunanya adalah `fedora` atau `ec2-user`.
- Untuk AMI RHEL, nama penggunanya adalah `ec2-user` atau `root`.
- Untuk AMI SUSE, nama penggunanya adalah `ec2-user` atau `root`.
- Untuk AMI Ubuntu, nama penggunanya adalah `ubuntu`.
- Untuk AMI Oracle, 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.

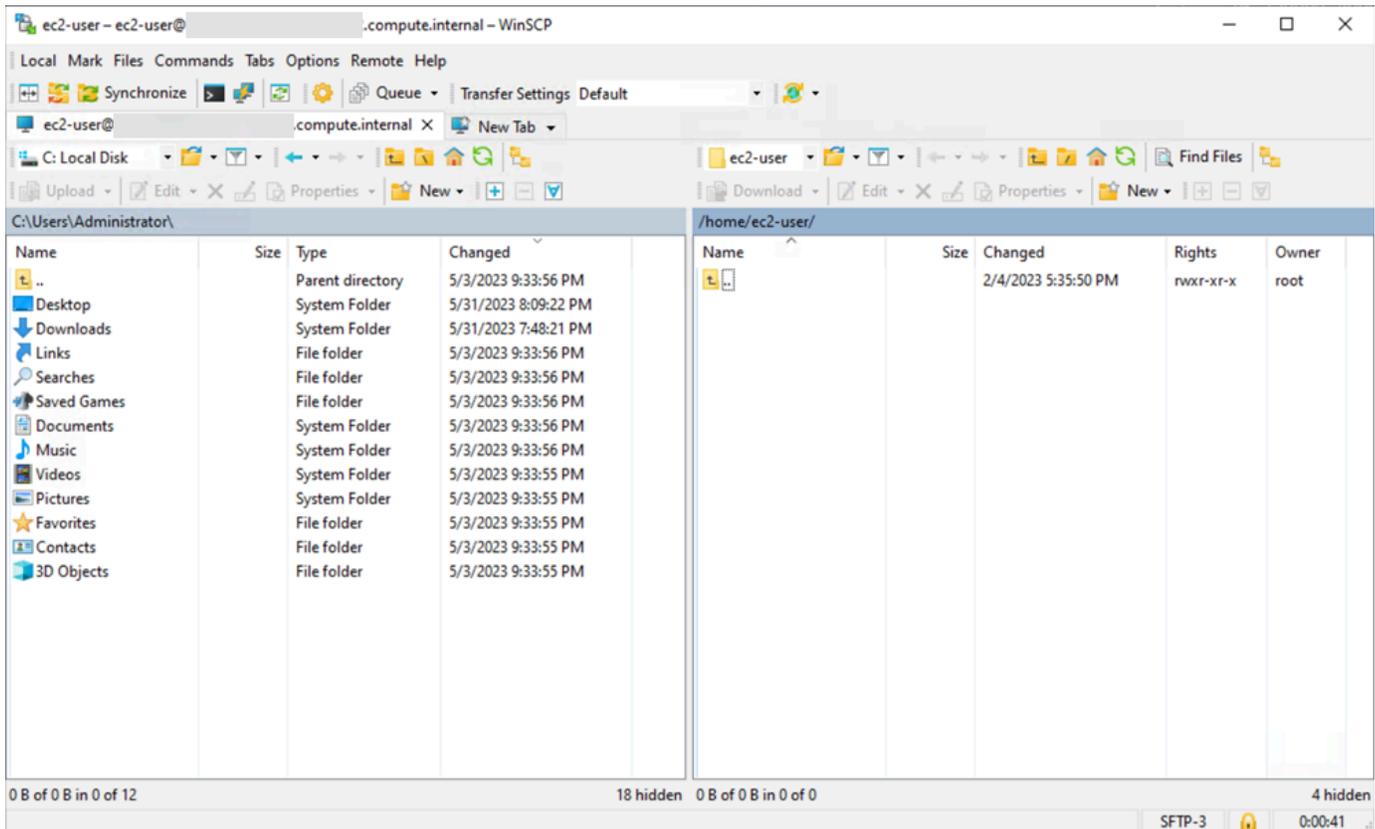
5. Tentukan file kunci pribadi untuk instans Anda.
 - a. Pilih tombol Lanjutan....
 - b. Di bawah SSH, pilih Autentikasi.
 - c. Tentukan jalur untuk file kunci privat Anda, atau pilih tombol ... untuk menjelajahi file pasangan kunci.
 - d. Pilih OKE.

Ini adalah tangkapan layar dari WinSCP versi 6.1:



WinSCP memerlukan file kunci privat PuTTY (.ppk). Anda dapat mengonversi file kunci keamanan .pem menjadi format .ppk menggunakan PuTTYgen. Untuk informasi selengkapnya, lihat [Konversi kunci privat Anda menggunakan PuTTYgen](#).

- (Opsional) Di panel kiri, pilih Direktori. Untuk Direktori jarak jauh, masukkan jalur direktori untuk menambahkan file. Untuk membuka pengaturan situs lanjutan untuk versi WinSCP yang lebih baru, pilih Lanjutan. Untuk menemukan pengaturan Direktori jarak jauh, pada Lingkungan, pilih Direktori.
- Pilih Login. Untuk menambahkan sidik jari host ke cache host, pilih Ya.



8. Setelah koneksi dibuat, di jendela koneksi, instans Linux Anda di sebelah kanan dan mesin lokal Anda di sebelah kiri. Anda dapat menyeret dan melepas file antara sistem file jarak jauh dan mesin lokal Anda. Untuk informasi lebih lanjut tentang WinSCP, lihat dokumentasi proyek di <http://winscp.net/eng/docs/start>.

Jika Anda menerima pesan kesalahan sehingga Anda tidak dapat menjalankan SCP untuk memulai transfer, pastikan bahwa Anda telah menginstal scp pada instans Linux.

Hubungkan ke instans Linux Anda dari Windows dengan Subsistem Windows untuk Linux (WSL)

Setelah meluncurkan instans, Anda dapat terhubung ke instans tersebut dan menggunakannya seperti Anda menggunakan komputer yang ada di depan Anda.

Petunjuk berikut menjelaskan cara terhubung ke instans Anda dengan distribusi Linux di Subsistem Windows untuk Linux (WSL). WSL dapat diunduh gratis dan memungkinkan Anda untuk menjalankan alat baris perintah Linux asli secara langsung di Windows, di samping desktop Windows tradisional Anda, tanpa overhead mesin virtual.

Dengan menginstal WSL, Anda dapat menggunakan lingkungan Linux native untuk terhubung ke instans EC2 Linux alih-alih menggunakan PuTTY atau PuTTYgen. Lingkungan Linux membuat

terhubung ke instans Linux Anda menjadi lebih mudah karena dilengkapi dengan klien SSH asli yang dapat Anda gunakan untuk terhubung ke instans Linux Anda dan mengubah izin file kunci.pem. Konsol Amazon EC2 menyediakan perintah SSH untuk menghubungkan ke instans Linux, dan Anda bisa mendapatkan keluaran verbose dari perintah SSH untuk pemecahan masalah. Untuk informasi selengkapnya, lihat [Subsistem Windows untuk Dokumentasi Linux](#).

Note

Setelah Anda menginstal WSL, semua prasyarat dan langkah-langkahnya sama seperti yang dijelaskan di [Hubungkan ke instans Linux dari Linux menggunakan SSH.](#), dan pengalamannya seperti menggunakan Linux native.

Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Daftar Isi

- [Prasyarat](#)
- [Hubungkan ke instans Linux Anda menggunakan WSL](#)
- [Mentransfer file ke instans Linux dari Linux menggunakan klien SCP](#)
- [Hapus instalasi WSL](#)

Prasyarat

Sebelum Anda terhubung ke instans Linux, selesaikan prasyarat berikut.

Verifikasi bahwa instans sudah siap

Setelah Anda meluncurkan instans, perlu beberapa menit sampai instans tersebut siap, sehingga Anda dapat terhubung dengannya. Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Anda dapat melihat informasi ini di kolom Pemeriksaan status di halaman Instans.

Verifikasi prasyarat umum untuk menghubungkan ke instans Anda

Untuk menemukan nama DNS publik atau alamat IP dari instans Anda dan nama pengguna yang harus Anda gunakan untuk terhubung ke instans Anda, lihat [Dapatkan informasi tentang instans Anda](#).

Instal Subsistem Windows untuk Linux (WSL) dan distribusi Linux di komputer lokal Anda

Instal WSL dan distribusi Linux menggunakan petunjuk di [Panduan Instalasi Windows 10](#).

Contoh dalam instruksi tersebut menginstal distribusi Ubuntu Linux, tetapi Anda dapat menginstal distribusi apa pun. Anda akan diminta untuk me-restart komputer Anda agar perubahan diterapkan.

Salin kunci privat dari Windows ke WSL

Di jendela terminal WSL, salin file `.pem` (untuk pasangan kunci yang Anda tentukan saat Anda meluncurkan instans) 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?](#). Untuk informasi selengkapnya tentang pasangan kunci dan instans Windows, lihat [pasangan kunci Amazon EC2 dan instans Windows](#).

```
cp /mnt/<Windows drive letter>/path/my-key-pair.pem ~/WSL-path/my-key-pair.pem
```

Hubungkan ke instans Linux Anda menggunakan WSL

Gunakan prosedur berikut untuk terhubung ke instans Linux Anda menggunakan Subsistem Windows untuk Linux (WSL). Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat [Memecahkan masalah saat menghubungkan ke instans Linux](#).

Untuk terhubung ke instans Anda menggunakan SSH

1. Di jendela terminal, gunakan `ssh` untuk terhubung ke instans. Anda menentukan jalur dan nama file kunci privat (`.pem`), nama pengguna untuk instans Anda, dan nama DNS publik atau alamat IPv6 untuk instans Anda. Untuk informasi selengkapnya tentang cara menemukan kunci privat, nama pengguna untuk instans Anda, dan nama DNS atau alamat IPv6 untuk sebuah instans, lihat [Temukan kunci pribadi dan atur izin](#) dan [Dapatkan informasi tentang instans Anda](#). Untuk terhubung ke instans Anda, gunakan salah satu perintah berikut.
 - (DNS Publik) Untuk terhubung menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
ssh -i /path/key-pair-name.pem instance-user-name@my-instance-public-dns-name
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, Anda dapat terhubung ke instans menggunakan alamat IPv6 instans tersebut. Tentukan perintah ssh dengan jalur ke file kunci privat (.pem), nama pengguna yang sesuai, dan alamat IPv6.

```
ssh -i /path/key-pair-name.pem instance-user-name@my-instance-IPv6-address
```

Anda akan melihat tanggapan 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)?
```

2. (Opsional) Pastikan sidik jari di 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.
3. Masukkan yes.

Anda akan melihat tanggapan seperti berikut:

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

Mentransfer file ke instans Linux dari Linux menggunakan klien SCP

Salah satu cara untuk mentransfer file antara komputer lokal dan instans Linux Anda adalah dengan menggunakan protokol salinan aman (SCP). Bagian ini menjelaskan cara mentransfer file dengan SCP. Prosedurnya mirip dengan prosedur untuk menghubungkan ke sebuah instans dengan SSH.

Prasyarat

- Verifikasi prasyarat umum untuk mentransfer file ke instans Anda.

Sebelum mentransfer file antara mesin lokal dan instans Anda, lakukan tindakan berikut untuk memastikan bahwa Anda memiliki semua informasi yang dibutuhkan.

- [Dapatkan informasi tentang instans Anda](#)
- [Temukan kunci pribadi dan atur izin](#)

- [\(Opsional\) Dapatkan sidik jari instans](#)
- 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 memandu Anda dalam menggunakan SCP untuk mentransfer file. Jika Anda sudah terhubung ke instans dengan SSH dan telah memverifikasi sidik jarinya, Anda dapat memulai dengan langkah yang berisi perintah SCP (langkah 4).

Untuk menggunakan SCP untuk mentransfer file

1. Transfer file ke instans Anda menggunakan nama DNS publik instans tersebut. Misalnya, jika nama file kunci privat adalah `key-pair-name`, maka file yang akan ditransfer adalah `SampleFile.txt`, nama pengguna adalah `instance-user-name`, dan nama DNS publik dari instans tersebut adalah `my-instance-public-dns-name`, atau alamat IPv6-nya adalah `my-instance-IPv6-address`. Gunakan salah satu perintah berikut untuk menyalin file ke direktori beranda `instance-user-name`.
 - (DNS Publik) Untuk mentransfer file menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
scp -i /path/key-pair-name.pem /path/SampleFile.txt instance-user-name@my-instance-public-dns-name:~
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, Anda dapat mentransfer file menggunakan alamat IPv6 instans tersebut. Alamat IPv6 harus diapit tanda kurung siku ([]), yang harus di-escape (\).

```
scp -i /path/key-pair-name.pem /path/SampleFile.txt instance-user-name@[my-instance-IPv6-address]:~
```

Anda akan melihat tanggapan 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)?
```

2. (Opsional) Pastikan sidik jari di 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.
3. Masukkan **yes**.

Anda akan melihat tanggapan seperti berikut:

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
Sending file modes: C0644 20 SampleFile.txt
Sink: C0644 20 SampleFile.txt
SampleFile.txt                               100%  20    0.0KB/s   00:00
```

Jika Anda menerima kesalahan "bash: scp: command not found", Anda harus menginstal scp terlebih dahulu pada instans Linux Anda. Untuk beberapa sistem operasi, letaknya adalah di paket `openssh-clients`. Untuk varian Amazon Linux, seperti AMI dengan pengoptimalan Amazon ECS, gunakan perintah berikut untuk menginstal scp:

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

4. Untuk mentransfer file ke arah lain (dari instans Amazon EC2 ke komputer lokal Anda), balikkan urutan parameter host. Misalnya, untuk mentransfer file `SampleFile.txt` dari instans EC2 Anda kembali ke direktori beranda di komputer lokal Anda sebagai `SampleFile2.txt`, gunakan salah satu dari perintah berikut di komputer lokal Anda.
 - (DNS Publik) Untuk mentransfer file menggunakan nama DNS publik instans Anda, masukkan perintah berikut.

```
scp -i /path/key-pair-name.pem instance-user-
name@ec2-198-51-100-1.compute-1.amazonaws.com:~/SampleFile.txt ~/
SampleFile2.txt
```

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk mentransfer file ke arah lain menggunakan alamat IPv6 instans tersebut, masukkan perintah berikut.

```
scp -i /path/key-pair-name.pem instance-user-name@
\[2001:db8:1234:1a00:9691:9503:25ad:1761\]:~/SampleFile.txt ~/SampleFile2.txt
```

Hapus instalasi WSL

Untuk informasi tentang membongkar Subsistem Windows untuk Linux, lihat [Bagaimana cara membongkar Distribusi WSL?](#).

Hubungkan ke instans Linux Anda dengan EC2 Instance Connect

Amazon EC2 Instance Connect memberikan cara sederhana dan aman untuk terhubung ke instans Linux Anda menggunakan Secure Shell (SSH). Dengan EC2 Instance Connect, Anda menggunakan [kebijakan](#) dan [prinsipal 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 terhubung ke instans Anda menggunakan konsol Amazon EC2 atau klien SSH pilihan Anda.

Saat Anda terhubung ke sebuah instans menggunakan EC2 Instance Connect, API Instance Connect mendorong kunci publik SSH ke [metadana instans](#) yang tetap bertahan selama 60 detik. Kebijakan IAM yang dilampirkan ke pengguna mengizinkan pengguna Anda untuk mendorong kunci publik ke metadana instans. Daemon SSH menggunakan `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, yang dikonfigurasi saat Instance Connect diinstal, untuk mencari kunci publik dari metadana instans untuk autentikasi dan menghubungkan Anda ke instans.

Anda dapat menggunakan EC2 Instance Connect untuk terhubung ke instans yang memiliki alamat IP privat atau publik. Untuk informasi selengkapnya, lihat [Hubungkan menggunakan 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](#).

Tip

EC2 Instance Connect adalah salah satu opsi untuk terhubung ke instans Linux Anda. Untuk opsi lain, lihat [Hubungkan ke instans Linux Anda](#). Untuk terhubung ke instance Windows, lihat [Hubungkan ke instans Windows Anda](#).

Daftar Isi

- [Tutorial: Selesaikan konfigurasi yang diperlukan untuk terhubung ke instans Anda menggunakan EC2 Instance Connect](#)
- [Prasyarat](#)
- [Berikan izin IAM untuk EC2 instans Connect](#)
- [Instal EC2 instans Connect pada instans EC2 Anda](#)
- [Hubungkan menggunakan EC2 Instance Connect](#)
- [Hapus instalasi EC2 Instance Connect](#)

Tutorial: Selesaikan konfigurasi yang diperlukan untuk terhubung 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 terhubung ke instans Anda. Tujuan dari tutorial ini adalah untuk memandu Anda melalui tugas-tugas untuk menyelesaikan konfigurasi prasyarat.

Ikhtisar tutorial

Dalam tutorial ini, Anda akan menyelesaikan empat tugas berikut:

- [Tugas 1: Membuat dan melampirkan kebijakan IAM untuk memungkinkan Anda menggunakan EC2 Instance Connect](#)

Pertama, Anda akan membuat kebijakan IAM yang berisi izin IAM yang memungkinkan Anda 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: Membuat grup keamanan untuk mengizinkan 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 instans Anda](#)

Anda kemudian akan meluncurkan instans 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: Connect ke instans](#)

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: Membuat dan melampirkan kebijakan IAM untuk memungkinkan Anda 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 pada identitas IAM Anda (pengguna, grup pengguna, atau peran) untuk memberi Anda izin yang diperlukan untuk mendorong kunci publik ke metadata instans.

Tujuan tugas

Dalam tugas ini, Anda akan membuat kebijakan IAM yang memberikan izin untuk mendorong kunci publik ke instance. Tindakan spesifik untuk memungkinkan adalah `ec2-instance-connect:SendSSHPublicKey`. Anda juga harus mengizinkan `ec2:DescribeInstances` tindakan 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:

```
{
  "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. Kami menggunakan kebijakan yang sangat permisif ini untuk menjaga tutorial tetap sederhana dan fokus pada konfigurasi spesifik yang diajarkan tutorial ini. [Namun, dalam lingkungan produksi, kami menyarankan agar kebijakan IAM Anda dikonfigurasi untuk memberikan izin hak istimewa paling sedikit.](#) Untuk contoh kebijakan IAM, lihat [Berikan izin IAM untuk EC2 instans Connect.](#)

Langkah-langkah untuk membuat dan melampirkan kebijakan IAM

Gunakan langkah-langkah berikut untuk membuat dan melampirkan kebijakan IAM. Untuk melihat animasi langkah-langkahnya, lihat [Melihat animasi: Buat kebijakan IAM](#) dan [Melihat animasi: Lampirkan kebijakan IAM](#).

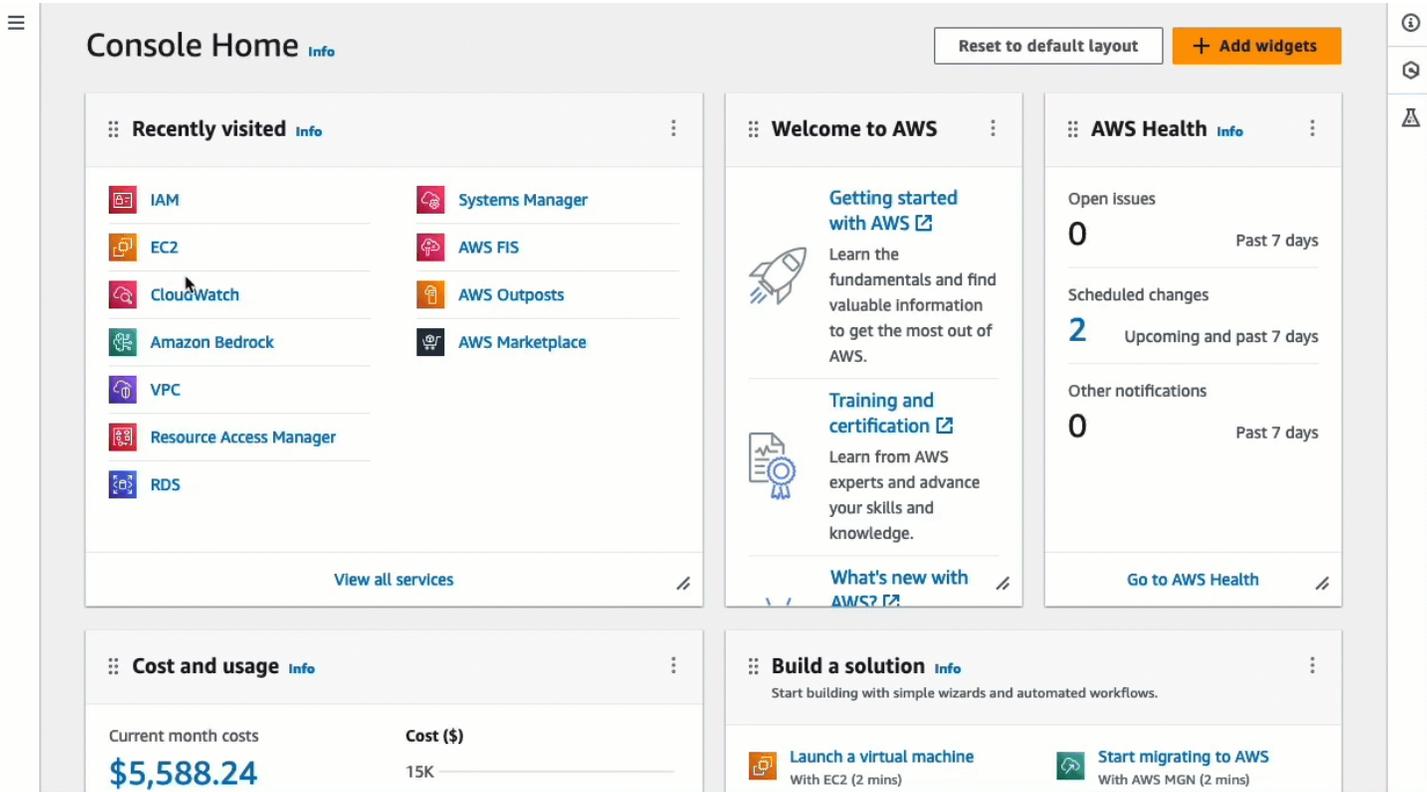
Untuk membuat dan melampirkan kebijakan IAM yang memungkinkan Anda menggunakan EC2 Instance Connect untuk terhubung ke instans

1. Pertama buat kebijakan IAM

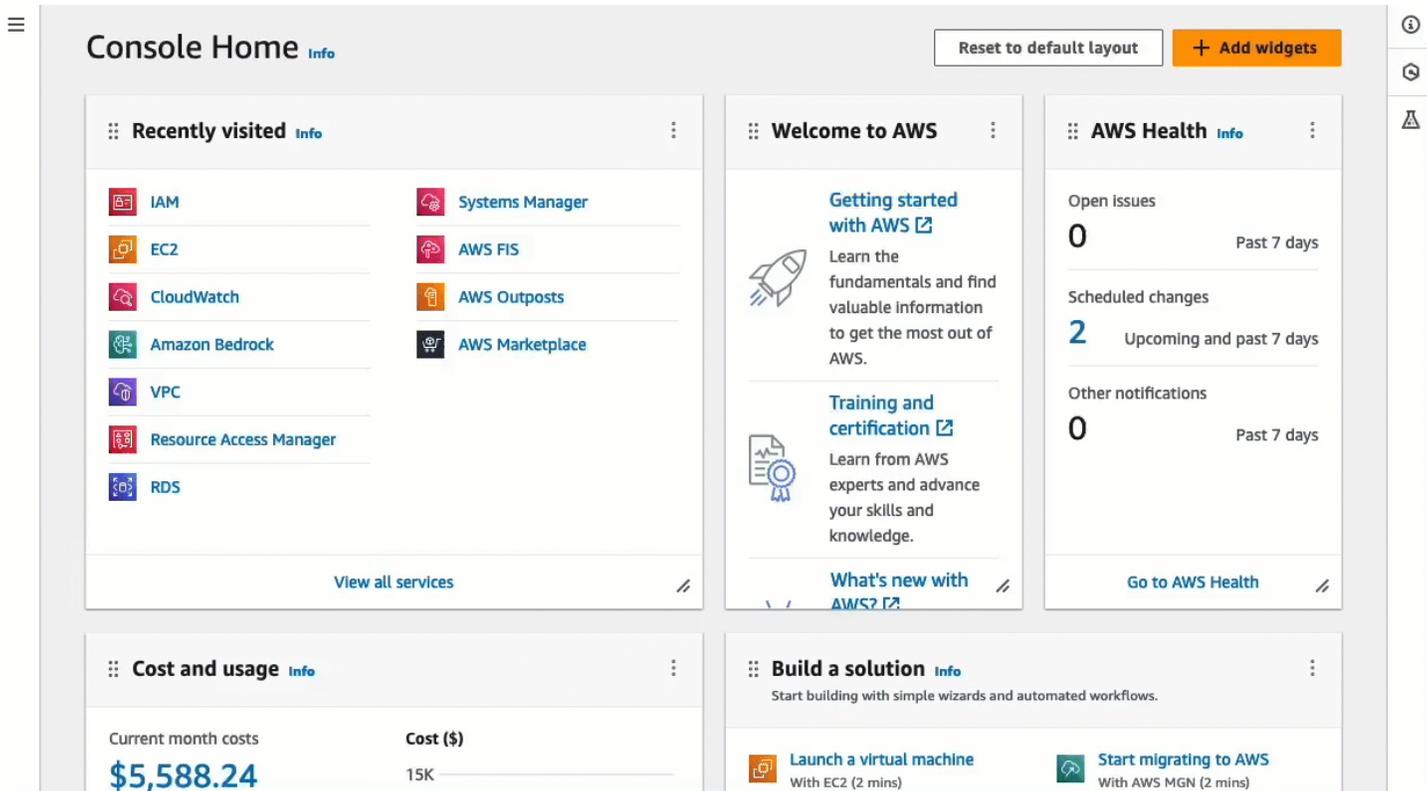
- a. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
- b. Di panel navigasi, pilih Kebijakan.
- c. Pilih Buat kebijakan.
- d. Pada halaman Tentukan izin, lakukan hal berikut:
 - i. Untuk Layanan, pilih EC2 Instance Connect.
 - ii. Di bawah Tindakan diizinkan, di bidang pencarian mulai mengetik **send** untuk menampilkan tindakan yang relevan, lalu pilih PublicKeySendSSH.
 - iii. Di bawah Sumber Daya, pilih Semua. Untuk lingkungan produksi, kami sarankan untuk menentukan instance dengan ARN-nya, tetapi untuk tutorial ini, Anda mengizinkan semua instance.

- iv. Pilih Tambahkan lebih banyak izin.
 - 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 Selanjutnya.
- e. Pada halaman Review dan create, 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 samping identitas Anda (pengguna, grup pengguna, atau peran). Anda dapat menggunakan kotak pencarian untuk memfilter daftar entitas.
 - e. Pilih Lampirkan kebijakan.

Melihat animasi: Buat kebijakan IAM



Melihat animasi: Lampirkan kebijakan IAM



Tugas 2: Membuat grup keamanan untuk mengizinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda

Saat Anda menggunakan EC2 Instance Connect 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 sebuah instans; dalam hal ini, Anda harus mengizinkan lalu lintas dari komputer lokal Anda ke instans Anda. Untuk mengizinkan lalu lintas dari layanan EC2 Instance Connect, Anda harus membuat grup keamanan yang memungkinkan lalu lintas SSH masuk dari rentang alamat IP untuk layanan EC2 Instance Connect.

Rentang alamat IP untuk AWS layanan tersedia di <https://ip-ranges.amazonaws.com/ip-ranges.json>. Rentang alamat IP EC2 Instance Connect diidentifikasi oleh "service": "EC2_INSTANCE_CONNECT".

Tujuan tugas

Dalam tugas ini, pertama Anda akan menemukan rentang alamat IP untuk EC2_INSTANCE_CONNECT Wilayah AWS di mana instance Anda berada. Kemudian Anda akan membuat grup keamanan yang memungkinkan lalu lintas SSH masuk pada port 22 dari rentang alamat IP tersebut.

Langkah-langkah untuk membuat grup keamanan

Gunakan langkah-langkah berikut untuk membuat grup keamanan. Untuk melihat animasi langkah-langkahnya, lihat [Lihat animasi: Dapatkan rentang alamat IP untuk EC2 Instance Connect untuk Wilayah tertentu](#) dan [Melihat animasi: Konfigurasi grup keamanan](#).

Untuk membuat grup keamanan yang memungkinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda

1. Pertama, dapatkan rentang alamat IP untuk layanan EC2 Instance Connect
 - a. Buka file JSON rentang alamat AWS IP di <https://ip-ranges.amazonaws.com/ip-ranges.json>.
 - b. Pilih Data Mentah.
 - c. Temukan rentang alamat IP EC2_INSTANCE_CONNECT untuk Wilayah AWS tempat instans Anda berada. Anda dapat menggunakan bidang pencarian browser untuk mencari layanan EC2_INSTANCE_CONNECT, dan terus mencari sampai Anda menemukan Wilayah di mana instans Anda berada.

Misalnya, jika instans Anda terletak di Wilayah AS Timur (Virginia N.) (us-east-1), rentang alamat IP untuk EC2_INSTANCE_CONNECT Wilayah tersebut adalah 18.206.107.24/29.

 Note

Rentang alamat IP berbeda untuk masing-masing Wilayah AWS.

- d. Salin rentang alamat IP yang muncul di sebelah `ip_prefix`. Anda akan menggunakan rentang alamat IP ini nanti dalam prosedur ini.

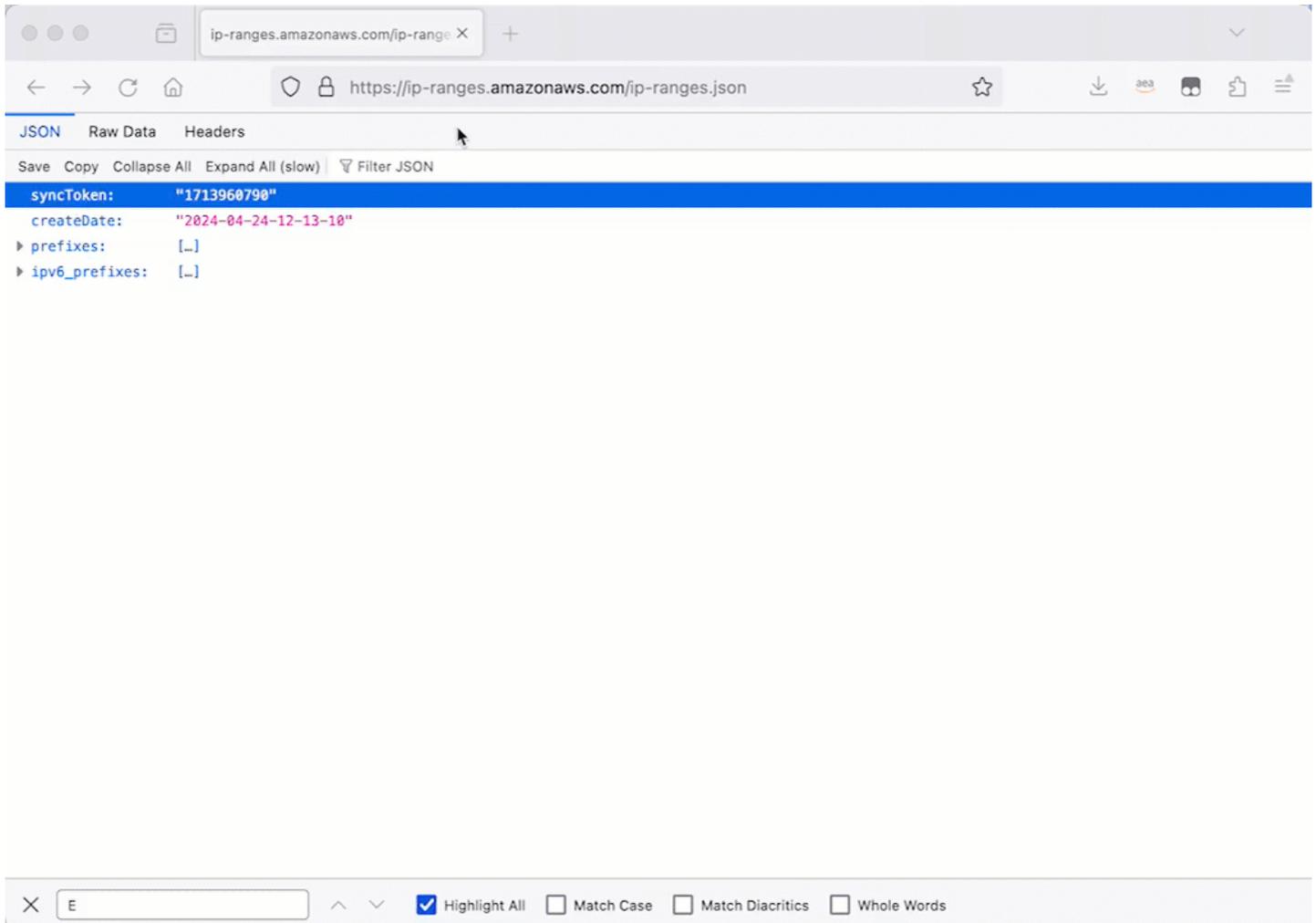
Untuk informasi selengkapnya tentang mengunduh file JSON rentang alamat AWS IP dan pemfilteran berdasarkan layanan, lihat [rentang alamat AWS IP](#) di Panduan Pengguna Amazon VPC.

2. Kemudian buat grup keamanan dengan aturan masuk untuk mengizinkan lalu lintas dari rentang alamat IP yang disalin
 - a. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
 - b. Pada panel navigasi, pilih Security Groups (Grup Keamanan).
 - c. Pilih Buat grup keamanan.
 - d. Di Detail dasar, lakukan hal berikut:
 - i. Untuk nama grup Keamanan, masukkan nama yang berarti untuk grup keamanan Anda.
 - ii. Untuk Deskripsi, masukkan deskripsi yang bermakna untuk grup keamanan Anda.
 - e. Di bawah aturan Inbound, lakukan hal berikut:
 - i. Pilih Tambahkan aturan.
 - ii. Untuk Tipe, pilih SSH.
 - iii. Untuk Sumber, tinggalkan Custom.
 - iv. Di bidang di sebelah Sumber, tempel rentang alamat IP untuk layanan EC2 Instance Connect yang Anda salin sebelumnya dalam prosedur ini.

Misalnya, jika instans Anda terletak di Wilayah AS Timur (Virginia N.) (us-east-1), tempelkan rentang alamat IP berikut ke dalam bidang: 18.206.107.24/29

- f. Pilih Buat grup keamanan.

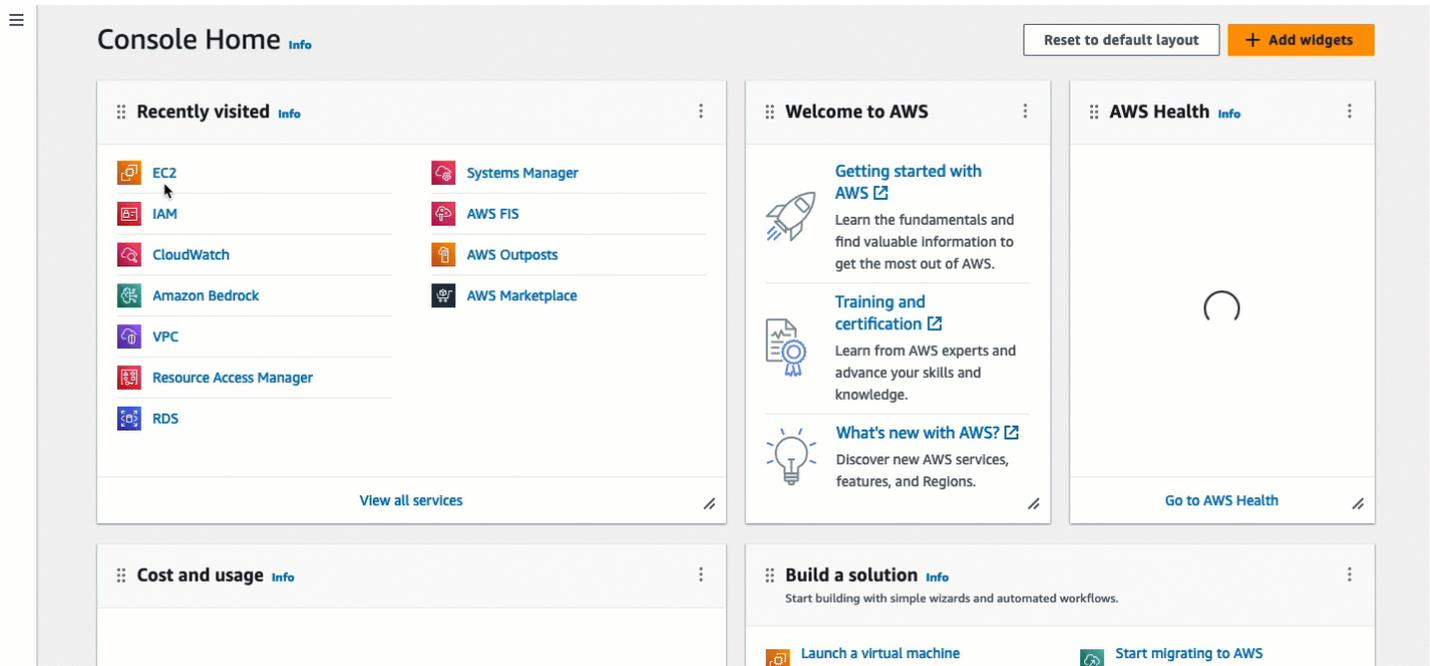
Lihat animasi: Dapatkan rentang alamat IP untuk EC2 Instance Connect untuk Wilayah tertentu



The screenshot shows a web browser displaying the IP ranges for a specific region. The browser address bar shows the URL `https://ip-ranges.amazonaws.com/ip-ranges.json`. The JSON data is displayed in the console, with the `syncToken` field highlighted in blue. The `createDate` field is also visible. Below the browser, a search bar is shown with the letter `E` entered, and search options like `Highlight All`, `Match Case`, `Match Diacritics`, and `Whole Words` are visible.

```
{
  "syncToken": "1713960790",
  "createDate": "2024-04-24-12-13-10",
  "prefixes": [],
  "ipv6_prefixes": []
}
```

Melihat animasi: Konfigurasi grup keamanan



Tugas 3: Luncurkan instans 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, kami menentukan AMI yang sudah diinstal sebelumnya dengan EC2 Instance Connect.

Jika Anda meluncurkan instans tanpa EC2 Instance Connect pra-instal, dan Anda ingin menggunakan EC2 Instance Connect untuk menyambung ke instans Anda, Anda harus melakukan langkah-langkah konfigurasi tambahan. Langkah-langkah ini berada di luar cakupan tutorial ini.

Tujuan tugas

Dalam tugas ini, Anda akan meluncurkan instance dengan Amazon Linux 2023 AMI, yang sudah diinstal sebelumnya dengan EC2 Instance Connect. Anda juga akan 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 terhubung ke instans Anda, yang mendorong kunci publik ke metadata instans Anda, Anda tidak perlu menentukan kunci SSH saat meluncurkan instance Anda. Tetapi Anda harus memastikan bahwa instans Anda memiliki alamat IPv4 publik karena menggunakan EC2 Instance Connect di konsol Amazon EC2 hanya mendukung penyambungan ke instans dengan alamat IPv4 publik.

Langkah-langkah untuk meluncurkan instans Anda

Gunakan langkah-langkah berikut untuk meluncurkan instans Anda. Untuk melihat animasi langkah-langkahnya, lihat [Melihat animasi: Luncurkan instance Anda](#).

Untuk meluncurkan instans yang dapat menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk koneksi

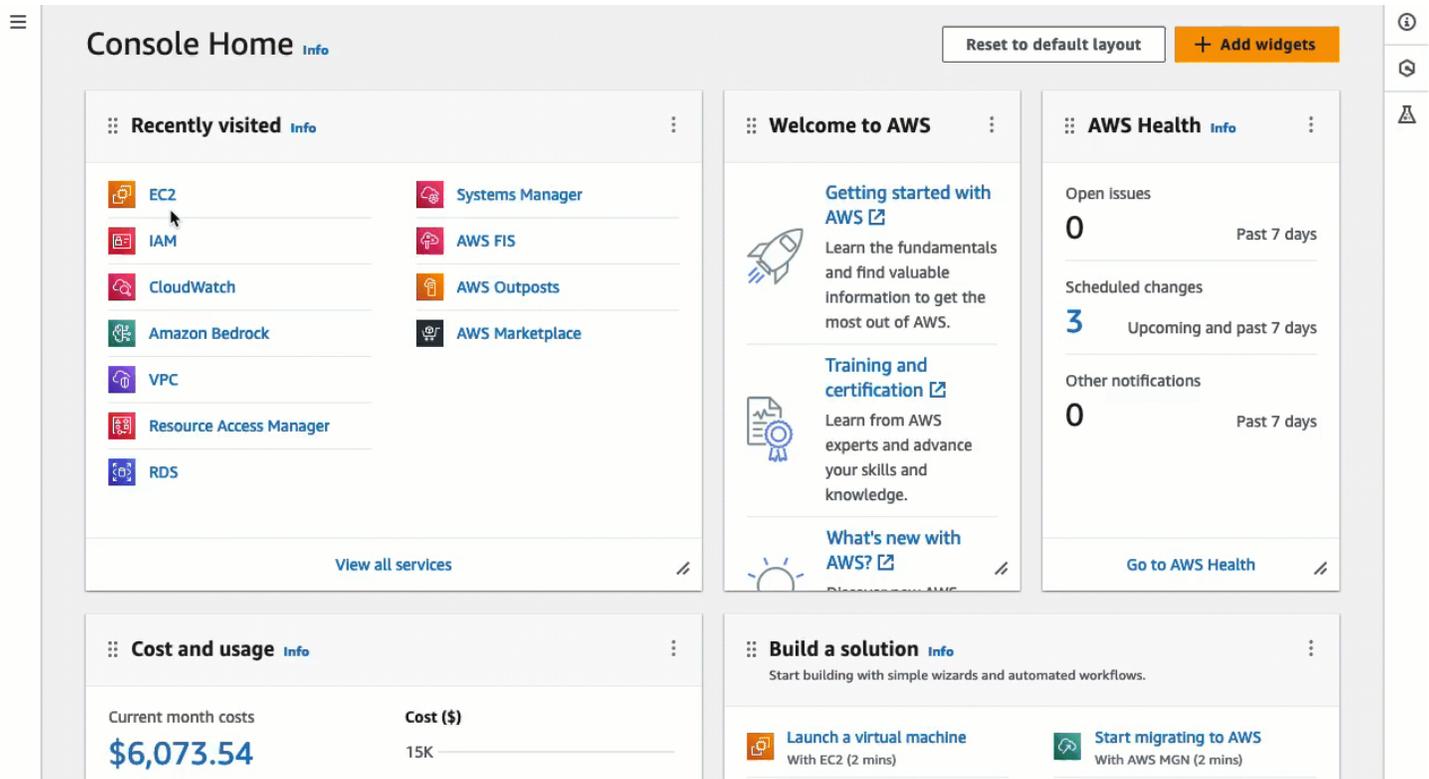
1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di bilah navigasi di bagian atas layar, AWS Wilayah saat ini ditampilkan (misalnya, Irlandia). Pilih Wilayah untuk meluncurkan instans Anda. Pilihan ini penting karena Anda membuat grup keamanan yang memungkinkan lalu lintas untuk 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 Gambar Aplikasi dan 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. Simpan pilihan default untuk tugas ini.
6. Di bawah Jenis Instance, untuk tipe Instance, pertahankan pilihan default, atau pilih jenis instans yang berbeda.
7. Di bawah Key pair (login), untuk nama Key pair, pilih Proceed without a key pair (Tidak disarankan). Saat Anda menggunakan EC2 Instance Connect untuk terhubung ke sebuah instans, EC2 Instance Connect mendorong key pair ke metadata instans, dan key pair inilah yang digunakan untuk koneksi.
8. Pada Pengaturan jaringan, lakukan hal berikut:
 - a. Untuk Auto-tetapkan IP publik, tinggalkan Aktifkan.

Note

Untuk menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk menyambung ke instans, instans harus memiliki alamat IPv4 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: Connect ke instans

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` 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 tugas prasyarat 1, 2, dan 3, koneksi harus berhasil.

Langkah-langkah untuk terhubung ke instans Anda

Gunakan langkah-langkah berikut untuk terhubung ke instans Anda. Untuk melihat animasi langkah-langkahnya, lihat [Melihat animasi: Connect ke instans Anda](#).

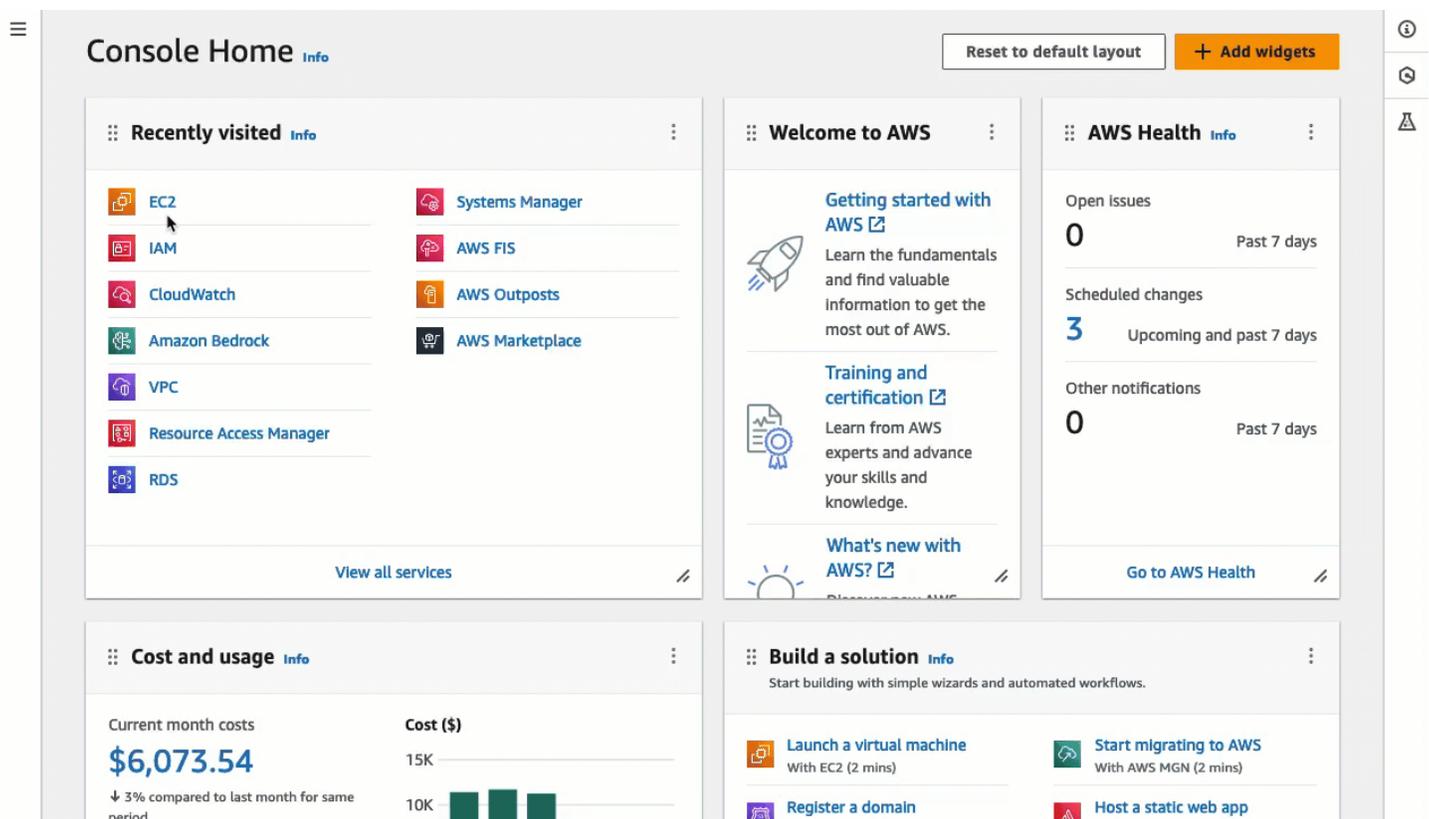
Untuk menghubungkan instans menggunakan EC2 Instance Connect di konsol Amazon EC2

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di bilah navigasi di bagian atas layar, AWS Wilayah saat ini ditampilkan (misalnya, Irlandia). Pilih Wilayah tempat instans Anda berada.
3. Di panel navigasi, pilih Instans.
4. Pilih instans Anda dan pilih Connect.
5. Pilih tab EC2 Instance Connect.
6. Untuk jenis Koneksi, pilih Connect menggunakan EC2 Instance Connect.
7. Pilih Hubungkan.

Jendela terminal terbuka di browser, dan Anda terhubung ke instance Anda.

Melihat animasi: Connect ke instans Anda



Prasyarat

Berikut ini adalah prasyarat untuk menginstal EC2 instans Connect dan untuk menggunakan EC2 instans Connect untuk terhubung ke sebuah instans:

- [Wilayah AWS](#)
- [Local Zones](#)

- [AMI](#)
- [Menginstal EC2 Instance Connect](#)
- [Alamat IPv4](#)
- [Akses jaringan](#)
- [Aturan grup keamanan](#)
- [Berikan izin](#)
- [Pengaturan komputer lokal](#)
- [nama pengguna](#)

Wilayah AWS

Didukung di semua Wilayah AWS kecuali Kanada Barat (Calgary).

Local Zones

Tidak didukung.

AMI

EC2 Instance Connect sudah diinstal sebelumnya pada AMI berikut:

- AL2023
- 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

EC2 Instance Connect tidak diinstal sebelumnya pada AMI berikut, tetapi Anda dapat menginstalnya pada instans yang diluncurkan menggunakan AMI berikut:

- Amazon Linux 2 sebelum versi 2.0.20190618
- 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 atau 18.04

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 AMI yang didukung, lihat bagian sebelumnya. Untuk instruksi instalasi, lihat [Instal EC2 instans Connect pada instans EC2 Anda](#).

Alamat IPv4

Instans Anda harus memiliki alamat IPv4 (baik privat maupun publik). EC2 Instance Connect tidak mendukung koneksi menggunakan alamat IPv6.

Akses 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 instans Anda harus memiliki alamat IP publik dan berada di subnet publik. Untuk informasi selengkapnya, lihat [Aktifkan akses internet](#) di Panduan Pengguna Amazon VPC.
- Jika pengguna Anda akan terhubung ke instans Anda melalui alamat IP pribadi instans, maka Anda harus membuat konektivitas jaringan pribadi ke VPC Anda, seperti dengan menggunakan AWS Direct Connect, AWS Site-to-Site VPN, atau mengintip VPC, sehingga pengguna Anda dapat mencapai alamat IP pribadi instans.

Jika instans Anda tidak memiliki alamat IPv4 publik dan Anda memilih untuk tidak mengonfigurasi akses jaringan seperti yang dijelaskan di atas, Anda dapat mempertimbangkan EC2 Instance Connect Endpoint sebagai alternatif dari EC2 Instance Connect. EC2 Instance Connect Endpoint memungkinkan Anda untuk terhubung ke instans melalui SSH atau RDP, tanpa mengharuskan instans memiliki alamat IPv4 publik. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Linux Anda menggunakan konsol Amazon EC2](#).

Aturan grup keamanan

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](#).

EC2 Instance Connect menggunakan rentang alamat IP tertentu untuk koneksi SSH berbasis peramban ke instans Anda (saat pengguna menggunakan konsol Amazon EC2 untuk terhubung ke sebuah instans). Jika pengguna Anda menggunakan konsol Amazon EC2 untuk terhubung ke sebuah instans, pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari rentang alamat IP untuk EC2_INSTANCE_CONNECT. Untuk mengidentifikasi rentang alamat, unduh file JSON yang disediakan oleh AWS dan filter untuk subset untuk EC2 Instance Connect, gunakan EC2_INSTANCE_CONNECT sebagai nilai layanan. Rentang alamat IP ini berbeda antara Wilayah AWS. Untuk informasi selengkapnya tentang mengunduh file JSON dan melakukan filter berdasarkan layanan, lihat [Rentang Alamat IP AWS](#) di Panduan Pengguna Amazon VPC.

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](#).

Pengaturan komputer lokal

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 instalasi 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 sebuah instans.

nama pengguna

Saat menggunakan EC2 Instance Connect untuk terhubung ke instans, nama pengguna harus memenuhi prasyarat 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 itu. `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.

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 `*`.

- Kondisi `ec2:osuser` memberikan izin untuk terhubung ke instans 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.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "ec2-instance-connect:SendSSHPublicKey",
    "Resource": [
      "arn:aws:ec2:region:account-id:instance/i-1234567890abcdef0",
      "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
    ],
    "Condition": {
      "StringEquals": {
        "ec2:osuser": "ami-username"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "ec2:DescribeInstances",
    "Resource": "*"
  }
]
```

Izinkan pengguna untuk terhubung ke instans dengan tanda tertentu

Attribute-based access control (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan tag yang dapat dilampirkan ke pengguna dan sumber daya. AWS Anda dapat menggunakan tanda sumber daya untuk mengontrol akses ke instans. Untuk informasi selengkapnya tentang penggunaan tag untuk mengontrol akses ke AWS sumber daya Anda, lihat [Mengontrol 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, 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.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "ec2-instance-connect:SendSSHPublicKey",
    "Resource": "arn:aws:ec2:region:account-id:instance/*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/tag-key": "tag-value"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "ec2:DescribeInstances",
    "Resource": "*"
  }
]
```

Instal EC2 instans Connect pada instans EC2 Anda

Untuk terhubung ke instans menggunakan EC2 Instance Connect, instans harus menginstal EC2 Instance Connect.

AMI berikut sudah diinstal sebelumnya dengan EC2 Instance Connect:

- 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

Jika instans Anda diluncurkan dengan salah satu AMI di daftar sebelumnya, Anda dapat melewati prosedur ini.

 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.

Prasyarat untuk menginstal Connect Instans EC2

- Luncurkan instans dengan salah satu AMI yang didukung berikut ini:

Amazon Linux 2 sebelum versi 2.0.20190618

AL2023 AMI minimal atau Amazon ECS AMI yang dioptimalkan

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

Jika instans Anda diluncurkan dengan versi Amazon Linux 2, macOS Sonoma, Ventura, atau Monterey, atau Ubuntu yang lebih baru, instans tersebut sudah diinstal sebelumnya dengan EC2 Instance Connect dan Anda dapat melewati prosedur ini.

- Verifikasi prasyarat umum untuk EC2 Instance Connect.

Untuk informasi selengkapnya, lihat [Prasyarat](#).

- Verifikasi prasyarat untuk menghubungkan ke instans Anda menggunakan klien SSH di mesin lokal Anda.

Jika mesin lokal Anda adalah Linux atau macOS, lihat [Hubungkan ke instans Linux dari Linux menggunakan SSH](#).. Jika mesin lokal Anda adalah Windows, lihat [Prasyarat](#).

Untuk informasi selengkapnya, lihat [Prasyarat koneksi SSH](#).

- Dapatkan ID instans.

Anda dapat memperoleh ID instans Anda menggunakan konsol Amazon EC2 (dari kolom ID instans). Jika mau, Anda dapat menggunakan [perintah describe-instance](#) (AWS CLI) atau [Get-EC2Instance\(\)](#).AWS Tools for Windows PowerShell

- Instal klien SSH di komputer lokal Anda.

Komputer lokal Anda kemungkinan besar telah menginstal klien SSH secara default. Anda dapat memeriksa klien SSH dengan mengetik ssh di baris perintah. Jika komputer lokal Anda tidak mengenali perintah tersebut, Anda dapat menginstal klien SSH. Untuk informasi tentang instalasi 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](#).

- (Ubuntu) Instal AWS CLI pada instance Anda.

Untuk menginstal EC2 Instance Connect pada instance Ubuntu, Anda harus menggunakan AWS CLI on the instance. Untuk informasi selengkapnya tentang [menginstal AWS CLI](#), lihat [Menginstal AWS CLI](#) di Panduan AWS Command Line Interface Pengguna.

Menginstal EC2 Instance Connect

Menginstal EC2 Instance Connect akan mengonfigurasi daemon SSH pada instans.

Gunakan salah satu prosedur berikut untuk menginstal EC2 Instance Connect tergantung 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 key pair SSH yang ditetapkan ke instans 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 dari Linux menggunakan 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 key pair SSH yang ditetapkan ke instans 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 dari Linux menggunakan 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 serta untuk tipe instans 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.rhel8.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.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.rhel8.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.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.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.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.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.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/ssh.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 key pair SSH yang ditetapkan ke instans 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 dari Linux menggunakan 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 [Perbarui sistem operasi dan perangkat lunak pada instance 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 key pair SSH yang ditetapkan ke instans 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 dari Linux menggunakan 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 serta tipe instans 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.rhel8.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.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.rhel8.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.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.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.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.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.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/ssh.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 key pair SSH yang ditetapkan ke instans Anda saat Anda meluncurkannya dan gunakan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk AMI Ubuntu, nama pengguna 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 dari Linux menggunakan 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) Pastikan bahwa 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.

Untuk informasi selengkapnya tentang paket EC2 Instance Connect, lihat [aws/aws-ec2](https://aws.amazon.com/ec2-instance-connect/) - di situs web. [instance-connect-config GitHub](https://github.com/aws/aws-ec2-instance-connect-config)

Hubungkan menggunakan EC2 Instance Connect

Petunjuk berikut menjelaskan cara menghubungkan ke instans Linux Anda menggunakan EC2 instans Connect.

Tentukan opsi koneksi mana yang akan digunakan. Opsi koneksi yang digunakan tergantung pada instans Anda memiliki alamat IPv4 publik:

- **Konsol Amazon EC2** — Untuk terhubung menggunakan konsol Amazon EC2, instans harus memiliki alamat IPv4 publik.
- **Klien SSH** — Jika instans tidak memiliki alamat IP publik, Anda dapat ter-connect ke instans melalui jaringan privat menggunakan klien SSH. Misalnya, Anda dapat terhubung dari dalam VPC yang sama atau melalui koneksi VPN, gateway transit, atau AWS Direct Connect

EC2 Instance Connect tidak mendukung koneksi menggunakan alamat IPv6.

Tip

EC2 Instance Connect adalah salah satu opsi untuk terhubung ke instans Linux Anda. Untuk opsi lain, lihat [Hubungkan ke instans Linux Anda](#). Untuk terhubung ke instance Windows, lihat [Hubungkan ke instans Windows Anda](#).

Opsi koneksi EC2 Instance Connect

- [Hubungkan menggunakan konsol Amazon EC2](#)
- [Hubungkan menggunakan kunci Anda sendiri dan klien SSH](#)
- [Connect menggunakan AWS CLI](#)
- [Pemecahan Masalah](#)

Hubungkan menggunakan konsol Amazon EC2

Anda dapat terhubung ke instans menggunakan konsol Amazon EC2 dengan memilih instans dari konsol dan memilih untuk terhubung menggunakan EC2 Instance Connect. Instance Connect menangani izin dan menyediakan koneksi yang berhasil.

Untuk terhubung menggunakan konsol Amazon EC2, instans harus memiliki alamat IPv4 publik. Sebelum menghubungkan, pastikan untuk meninjau semua [prasyarat](#).

Untuk terhubung ke instans Anda menggunakan klien berbasis browser dari 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. Untuk jenis Koneksi, pilih Connect menggunakan EC2 Instance Connect.
6. Untuk Nama Pengguna, verifikasi nama pengguna.
7. Pilih Hubungkan untuk membuka jendela terminal.

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. Hal ini memungkinkan Anda untuk memanfaatkan kemampuan instans 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](#).
- Ketika terhubung ke sebuah instans yang hanya memiliki alamat IP privat, komputer lokal yang Anda gunakan untuk 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 privat instans untuk membuat sesi SSH. Titik akhir layanan EC2 Instance Connect dapat dijangkau melalui internet atau melalui antarmuka virtual publik AWS Direct Connect . Untuk terhubung ke alamat IP privat instans, Anda dapat memanfaatkan layanan seperti [AWS Direct Connect](#), [AWS Site-to-Site VPN](#), atau [VPC peering](#).

Sebelum menghubungkan, pastikan untuk meninjau semua [prasyarat](#).

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 instans 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 Anda, dan nama DNS publik instans (jika menghubungkan 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
```

Connect menggunakan AWS CLI

Jika Anda mengetahui ID instans Anda, Anda dapat menggunakan AWS CLI perintah [ec2-instance-connect untuk terhubung](#) ke instans Anda menggunakan klien SSH. Jika Anda tidak menentukan tipe koneksi, EC2 instans Connect secara otomatis mencoba menyambung ke alamat IPv4 publik instans Anda. Jika instans Anda tidak memiliki alamat IPv4 publik, EC2 Instance Connect kemudian mencoba untuk menyambung ke alamat IPv4 privat instans Anda melalui [EC2 Instance Connect Endpoint](#). Jika instans Anda tidak memiliki alamat IPv4 privat atau VPC Anda tidak memiliki EC2 Instance Connect Endpoint, maka EC2 Instance Connect mencoba menyambung ke alamat IPv6 instans Anda.

Important

Sebelum Anda terhubung dengan metode ini, pastikan bahwa Anda telah mengonfigurasi AWS CLI, termasuk kredensial yang digunakannya, dan bahwa Anda menggunakan versi terbaru. AWS CLI Untuk informasi selengkapnya, lihat [Menginstal atau memperbarui versi terbaru AWS CLI](#) dan [Mengonfigurasi AWS CLI](#) dalam Panduan Pengguna AWS Command Line Interface .

Tipe koneksi

auto (default)

CLI mencoba terhubung menggunakan alamat IP instans dalam urutan berikut dan dengan jenis koneksi yang sesuai:

- IPv4 publik: `direct`
- IPv4 privat: `eice`
- IPv6: `direct`

`direct`

CLI mencoba untuk terhubung menggunakan alamat IP instans dalam urutan berikut (tidak terhubung melalui EC2 Instance Connect Endpoint):

- IPv4 Publik
- IPv6
- IPv4 privat

`eice`

CLI selalu menggunakan alamat IPv4 privat instans.

Note

Di masa depan, kita mungkin mengubah perilaku jenis auto koneksi. Untuk memastikan bahwa tipe koneksi yang Anda inginkan digunakan, kami sarankan Anda secara eksplisit mengatur `--connection-type` ke salah satu dari `direct` atau `eice`.

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.

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 instance Anda, gunakan perintah [ec2-instance-connect](#) CLI dan tentukan parameter dan ID instance. `ssh`

```
aws ec2-instance-connect ssh --instance-id i-1234567890example
```

Tip

Jika Anda mendapatkan kesalahan saat menggunakan perintah ini, pastikan Anda menggunakan AWS CLI versi 2. `sshParameter` ini hanya tersedia di AWS CLI versi 2. Untuk informasi selengkapnya, lihat [Tentang AWS CLI versi 2](#) di Panduan AWS Command Line Interface Pengguna.

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
```

Pemecahan Masalah

Jika Anda menerima kesalahan saat mencoba untuk terhubung ke instans Anda, lihat hal berikut:

- [Memecahkan masalah saat menghubungkan ke instans Linux](#)
- [Bagaimana cara memecahkan masalah menghubungkan ke instans EC2 saya menggunakan EC2 instans Connect?](#)

Hapus instalasi EC2 Instance Connect

Untuk menonaktifkan EC2 Instance Connect, hubungkan ke instans Anda dan hapus instalasi paket `ec2-instance-connect` yang diinstal pada OS. Jika konfigurasi `sshd` cocok dengan pengaturan saat Anda menginstal EC2 Instance Connect, menghapus instalasi `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 instalasi EC2 Instance Connect pada instans yang diluncurkan dengan Amazon Linux 2

1. Terhubung ke instans Anda menggunakan SSH. Tentukan key pair SSH yang Anda gunakan untuk instans saat meluncurkannya dan nama pengguna default untuk AL2023 atau Amazon Linux 2 AMI, 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 instalasi EC2 Instance Connect pada instans yang diluncurkan dengan AMI Ubuntu

1. Hubungkan ke instans Anda menggunakan SSH. Tentukan key pair SSH yang Anda gunakan untuk instans Anda saat Anda meluncurkannya dan nama pengguna default untuk AMI Ubuntu, yaitu `ubuntu`.

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
```

Hubungkan ke instans Windows Anda

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 instans Anda untuk memecahkan masalah boot, konfigurasi jaringan, dan masalah lainnya untuk instans yang dibangun di [Sistem AWS Nitro](#), Anda dapat menggunakan [Konsol Serial EC2 untuk instans Amazon EC2](#)

Daftar Isi

- [Connect ke instans Windows Anda menggunakan klien RDP](#)
- [Hubungkan ke instans Windows Anda menggunakan Fleet Manager](#)
- [Konfigurasi akun Anda](#)
- [Transfer file ke instans Windows](#)

Connect ke instans Windows Anda menggunakan klien RDP

Bagian berikut merinci prasyarat dan proses untuk terhubung ke instans Anda menggunakan alamat IPv4 atau IPv6 dengan klien RDP.

Prasyarat

Anda harus memenuhi prasyarat berikut untuk terhubung ke instance Windows Anda menggunakan klien RDP.

- 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, lihat [beranda Windows](#) dan cari unduhan untuk aplikasi Microsoft Remote Desktop.
 - (macOS X) Unduh [aplikasi Microsoft Remote Desktop](#) dari Mac App Store.
 - (Linux) Gunakan [Remmina](#).
- Temukan kunci privat

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

[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 \[Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa\]\(#\). Untuk langkah-langkah mengatur ulang kata sandi menggunakan dokumen Systems Manager, lihat \[Atur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna AWS Systems Manager\]\(#\).](#)

- Aktifkan lalu lintas RDP masuk dari alamat IP ke instans Anda

Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas RDP masuk (port 3389) dari alamat IP Anda. Grup keamanan default tidak mengizinkan lalu lintas RDP masuk secara default. Untuk informasi selengkapnya, lihat [Aturan-aturan untuk terhubung ke instans dari komputer Anda](#).

Tip

Anda dapat membuat [EC2 Instance Connect Endpoint](#) untuk terhubung ke instans Windows Anda menggunakan RDP tanpa alamat IPv4 publik.

Connect ke instance Windows menggunakan RDP dan alamat IPv4-nya

Untuk terhubung ke instance Windows, Anda harus mengambil kata sandi administrator awal dan menggunakan kata sandi ini saat Anda terhubung ke instans Anda menggunakan Remote Desktop. Diperlukan beberapa menit setelah peluncuran instans sebelum sandi ini tersedia.

Nama pengguna default untuk akun Administrator tergantung pada bahasa sistem operasi (OS) yang terkandung dalam AMI. Untuk memastikan nama pengguna yang benar, identifikasi bahasa OS AMI Anda, lalu pilih nama pengguna yang sesuai. Misalnya, untuk OS bahasa Inggris, nama

pengguna 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 pengguna Administrator (Other). Untuk informasi selengkapnya, lihat [Nama Lokal untuk Akun Administrator di Windows](#) di Microsoft TechNet Wiki.

Jika Anda telah menggabungkan instans Anda ke suatu domain, Anda dapat ter-connect ke instans Anda menggunakan kredensial domain yang telah Anda tentukan di AWS Directory Service. Pada layar login Remote Desktop, alih-alih menggunakan nama komputer lokal dan kata sandi yang dibuat, gunakan nama pengguna yang memenuhi syarat untuk administrator (misalnya **corp.example.com\Admin**) dan kata sandi untuk akun ini.

Jika Anda menerima kesalahan saat mencoba untuk terhubung ke instance Anda, lihat [the section called "Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh"](#).

Untuk terhubung ke instans Windows menggunakan klien RDP

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 instance 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 kunci pribadi (.pem) yang Anda tentukan saat meluncurkan instance. Pilih file dan pilih Buka untuk menyalin seluruh isi file ke jendela ini.
 - b. Pilih Dekripsi kata sandi. Halaman Dapatkan kata sandi Windows ditutup, dan kata sandi administrator default untuk instance muncul di bawah Kata Sandi, 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.
8. Pilih Unduh file desktop jarak jauh. Setelah selesai mengunduh file, pilih Batalkan untuk kembali ke halaman Instans. Arahkan ke direktori unduhan Anda, dan buka file RDP.

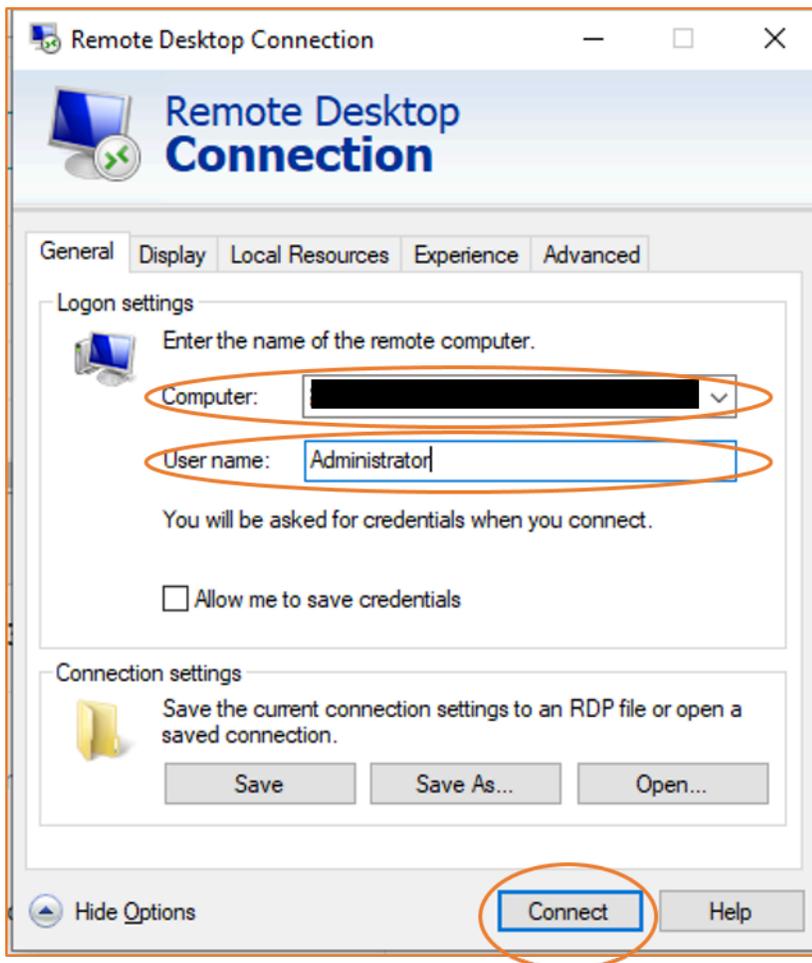
9. Anda mungkin mendapatkan peringatan bahwa penerbit koneksi jarak jauh tidak dikenal. Pilih Hubungkan untuk terus terhubung ke instans Anda.
10. Akun administrator dipilih secara default. Tempel kata sandi yang Anda salin sebelumnya, lalu pilih OK.
11. Karena sifat dari sertifikat yang ditandatangani sendiri, Anda mungkin mendapatkan peringatan bahwa sertifikat keamanan tidak dapat diautentikasi. Lakukan salah satu hal berikut ini:
 - Jika Anda mempercayai sertifikat, pilih Ya untuk terhubung ke instans Anda.
 - [Windows] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai dalam 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 Actions, Monitor dan troubleshoot, Dapatkan log sistem.
 - [Mac OS X] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai dalam log sistem untuk mengonfirmasi identitas komputer jarak jauh. Pilih Tampilkan Sertifikat, perluas Detail, dan pilih Sidik Jari SHA1. Bandingkan nilai ini dengan nilai RDPCERTIFICATE-THUMBPRINT di Actions, Monitor dan troubleshoot, Dapatkan log sistem.

Connect ke instance Windows menggunakan RDP dan alamat IPv6-nya

Jika Anda telah [mengaktifkan VPC Anda untuk IPv6](#) dan [menetapkan alamat IPv6 ke instans Windows Anda](#), maka Anda dapat menggunakan klien RDP untuk menghubungkannya ke instans Anda menggunakan alamat IPv6 (misalnya, 2001:db8:1234:1a00:9691:9503:25ad:1761), bukan menggunakan alamat IPv4 publik atau nama host DNS publik.

Untuk menghubungkan ke instans Windows Anda menggunakan alamat IPv6

1. Dapatkan kata sandi administrator awal untuk instans Anda, seperti yang dijelaskan di [Connect ke instans Windows Anda menggunakan klien RDP](#). Kata sandi ini diperlukan untuk terhubung ke instans Anda.
2. (Windows) Buka klien RDP di komputer Windows Anda, pilih Tampilkan Opsi, dan lakukan hal berikut:



- Untuk Komputer, masukkan alamat IPv6 instans Windows Anda.
- Untuk Nama pengguna, masukkan Administrator.
- Pilih Hubungkan.
- Saat diminta, masukkan kata sandi yang Anda simpan sebelumnya.

(macOS X) Buka klien RDP di komputer Anda dan lakukan hal berikut:

- Pilih Baru.
 - Untuk Nama PC, masukkan alamat IPv6 instans Windows Anda.
 - Untuk Nama pengguna, masukkan Administrator.
 - Tutup kotak dialog. Di bawah Desktop Saya, pilih koneksi, dan pilih Mulai.
 - Saat diminta, masukkan kata sandi yang Anda simpan sebelumnya.
3. Karena sifat dari sertifikat yang ditandatangani sendiri, Anda mungkin mendapatkan peringatan bahwa sertifikat keamanan tidak dapat diautentikasi. Jika Anda mempercayai sertifikat tersebut,

Anda dapat memilih Ya atau Lanjutkan. Jika tidak, Anda dapat memverifikasi identitas komputer jarak jauh, seperti yang dijelaskan di [Connect ke instans Windows Anda menggunakan klien RDP](#).

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 instance Windows pada halaman yang sama di halaman yang sama. AWS Management Console 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 [Hubungkan ke simpul terkelola menggunakan Remote Desktop](#) di Panduan AWS Systems Manager Pengguna.

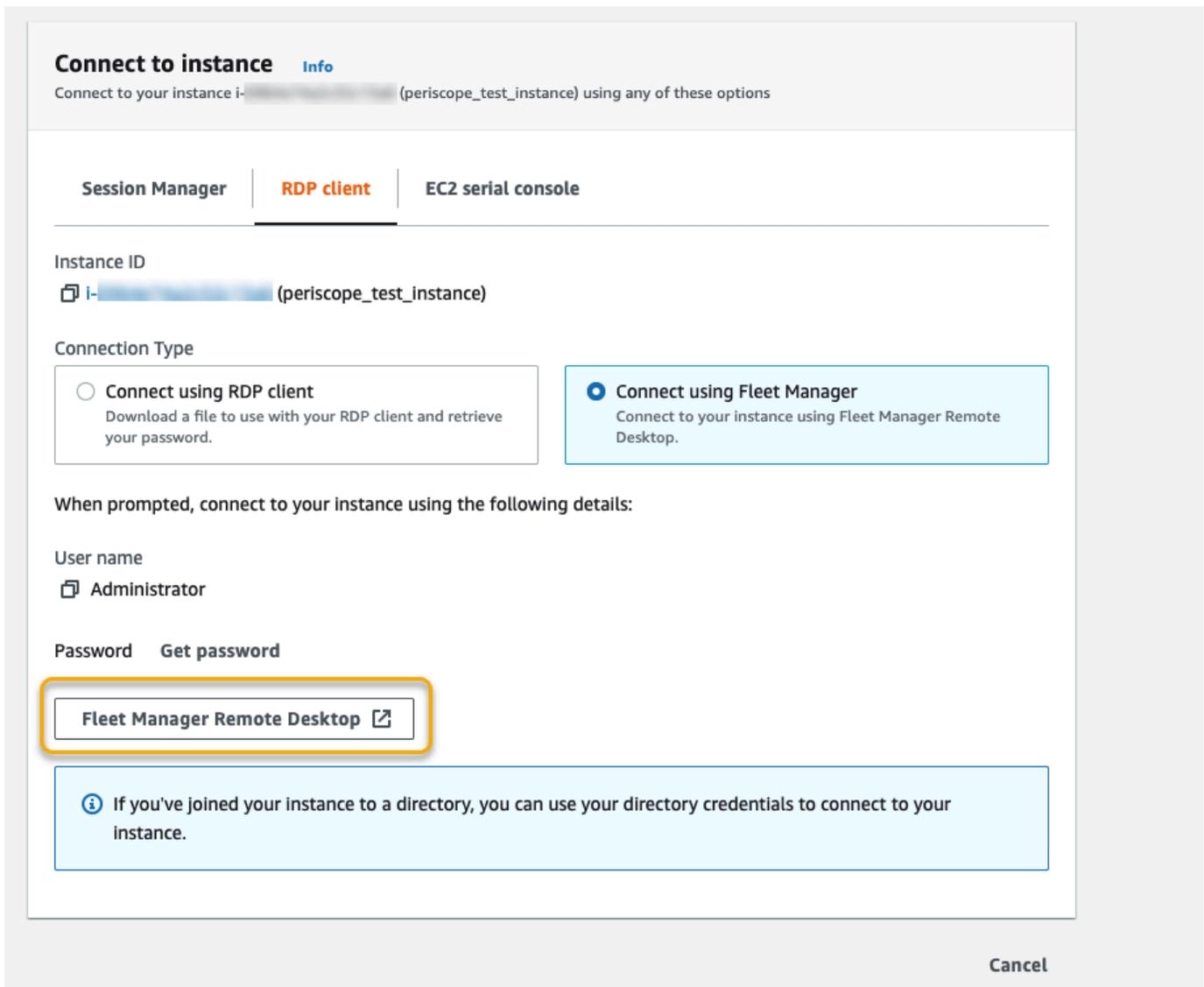
Sebelum mencoba menghubungkan ke sebuah instans menggunakan Fleet Manager, pastikan bahwa langkah-langkah penyiapan yang diperlukan telah diselesaikan. Untuk informasi selengkapnya, lihat [Menyiapkan Fleet Manager](#).

Note

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.

Untuk menghubungkan ke instans menggunakan RDP dengan Fleet Manager (konsol)

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 Hubungkan ke instans, pilih opsi Hubungkan menggunakan Fleet Manager, lalu pilih Desktop Jarak Jauh Fleet Manager. Halaman Desktop Jarak Jauh Fleet Manager akan terbuka di konsol AWS Systems Manager .



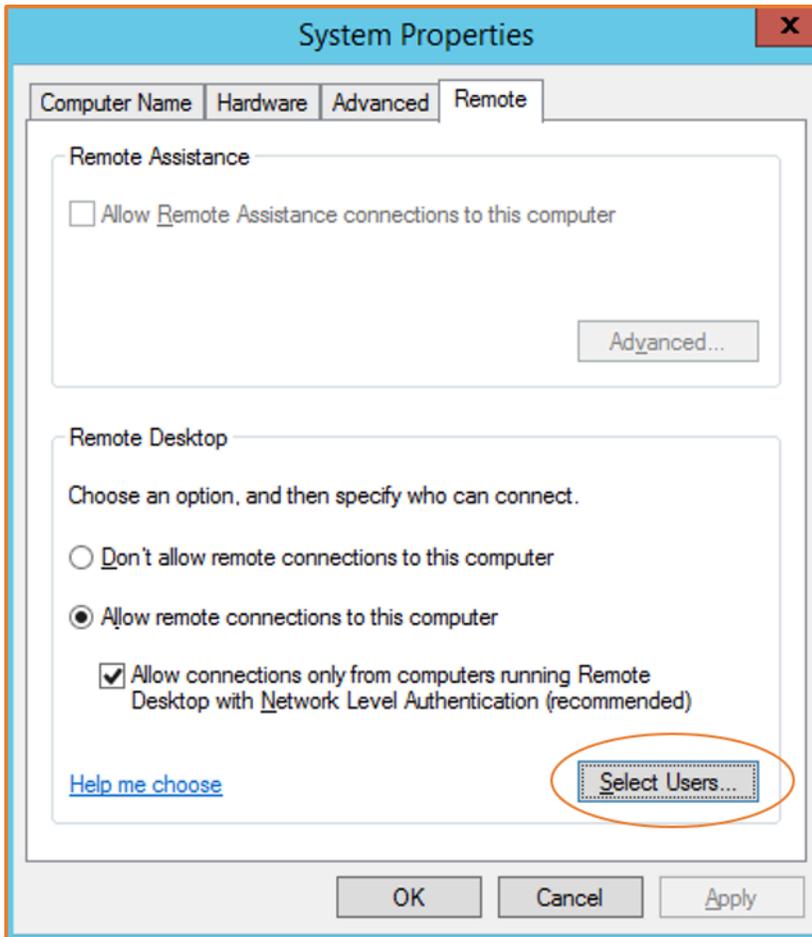
Untuk informasi selengkapnya tentang menghubungkan ke instans Windows dari halaman Desktop Jarak Jauh Fleet Manager, lihat [Hubungkan menggunakan Desktop Jarak Jauh](#) di Panduan Pengguna AWS Systems Manager .

Konfigurasi akun Anda

Setelah Anda terhubung melalui RDP, kami sarankan Anda melakukan hal 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.



Transfer file ke instans Windows

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 dari 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.

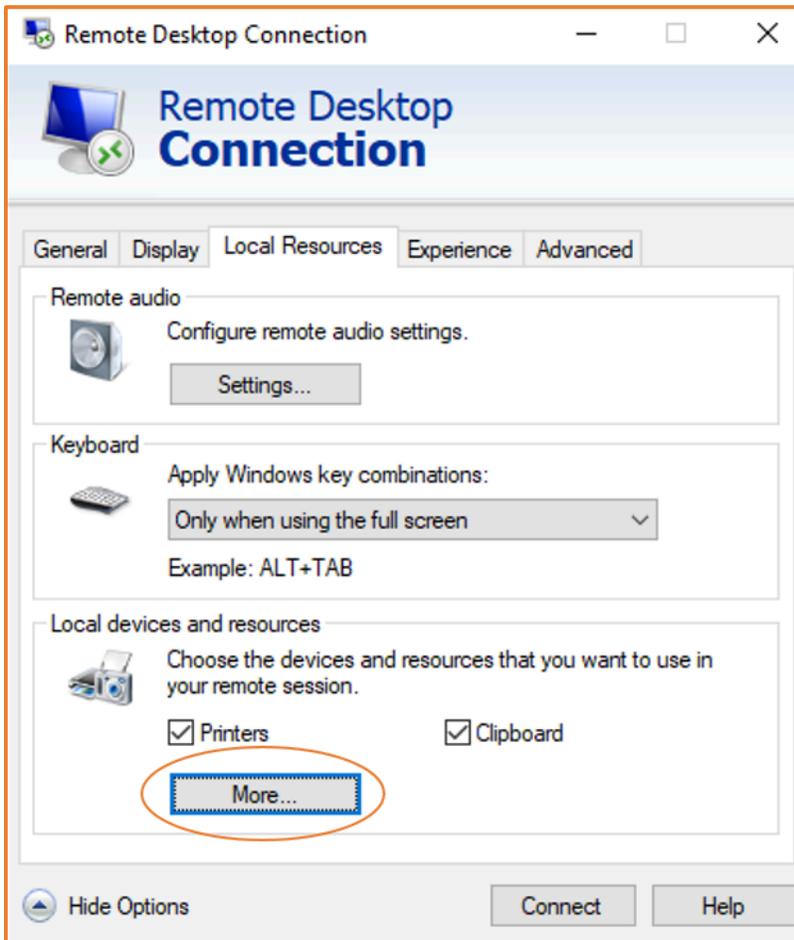
Untuk mengakses file lokal Anda dari instans Windows, Anda harus mengaktifkan fitur berbagi file lokal dengan memetakan drive sesi jarak jauh ke drive lokal Anda. Langkah-langkahnya sedikit

berbeda tergantung pada apakah sistem operasi komputer lokal Anda adalah Windows atau macOS X.

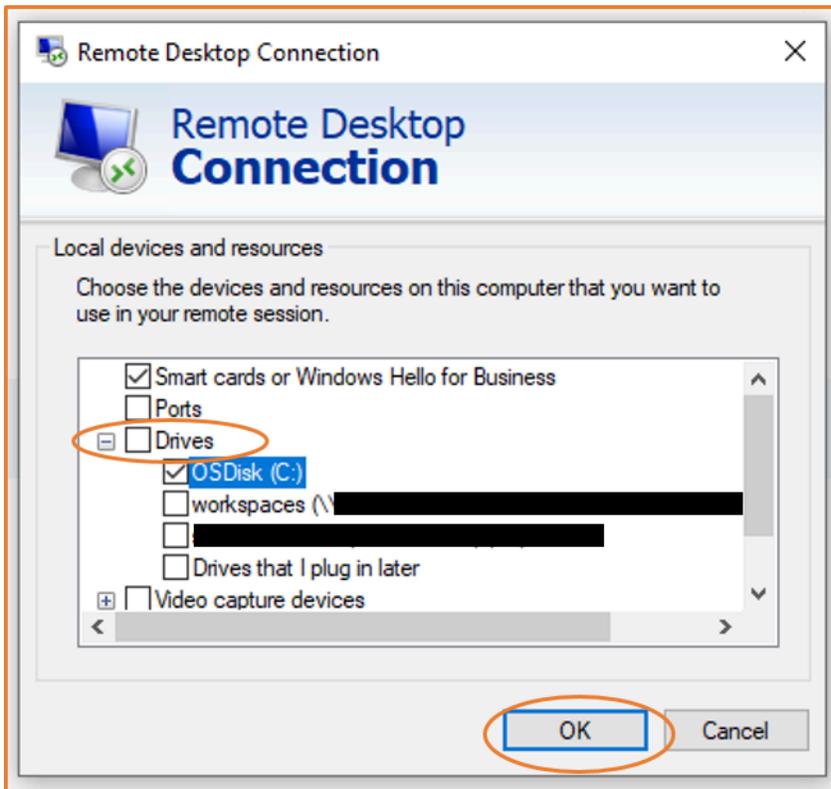
Windows

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 instans ke bidang Komputer 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.

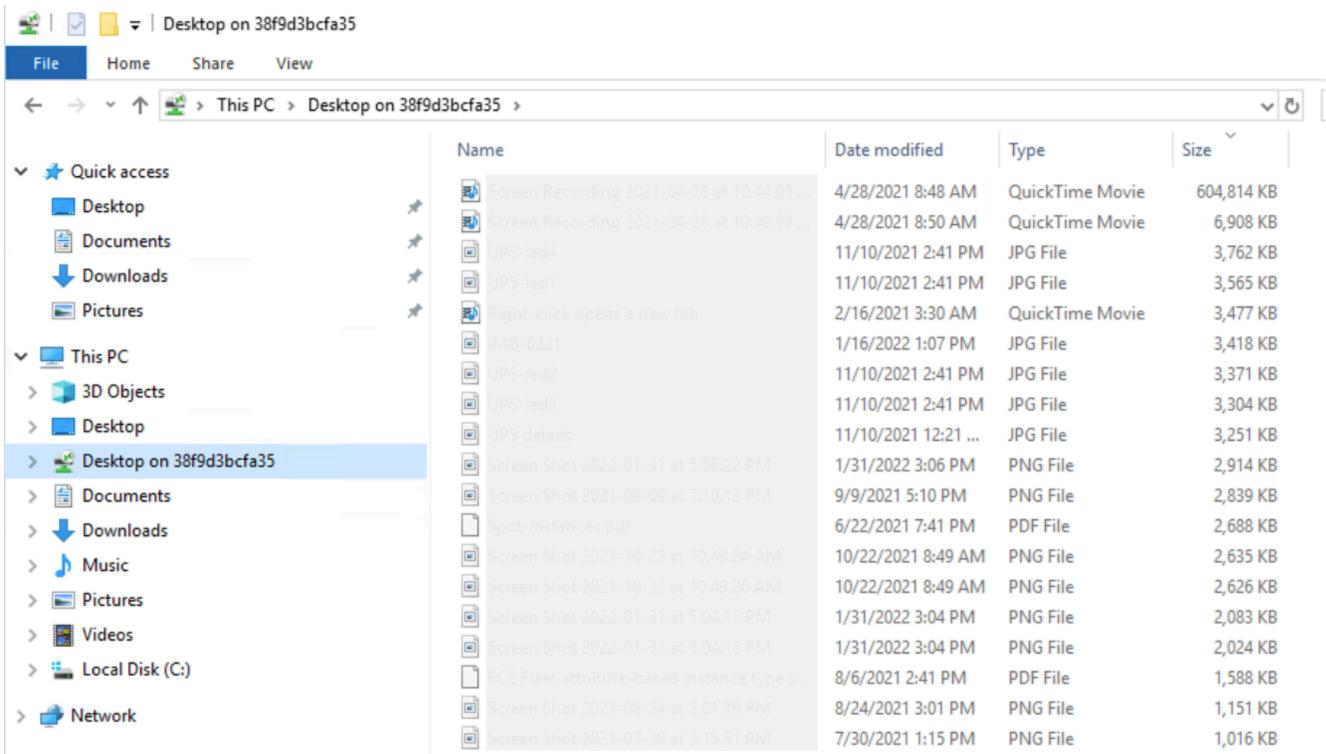


8. Pilih Hubungkan untuk terhubung ke instans Windows Anda.

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. Klik kanan file RDP, dan pilih Edit.
4. Pilih tab Folder, dan pilih kotak centang Alihkan folder.



Untuk informasi selengkapnya tentang membuat perangkat lokal tersedia untuk sesi jarak jauh di komputer Mac, lihat [Memulai dengan klien macOS](#).

Terhubung menggunakan Session Manager

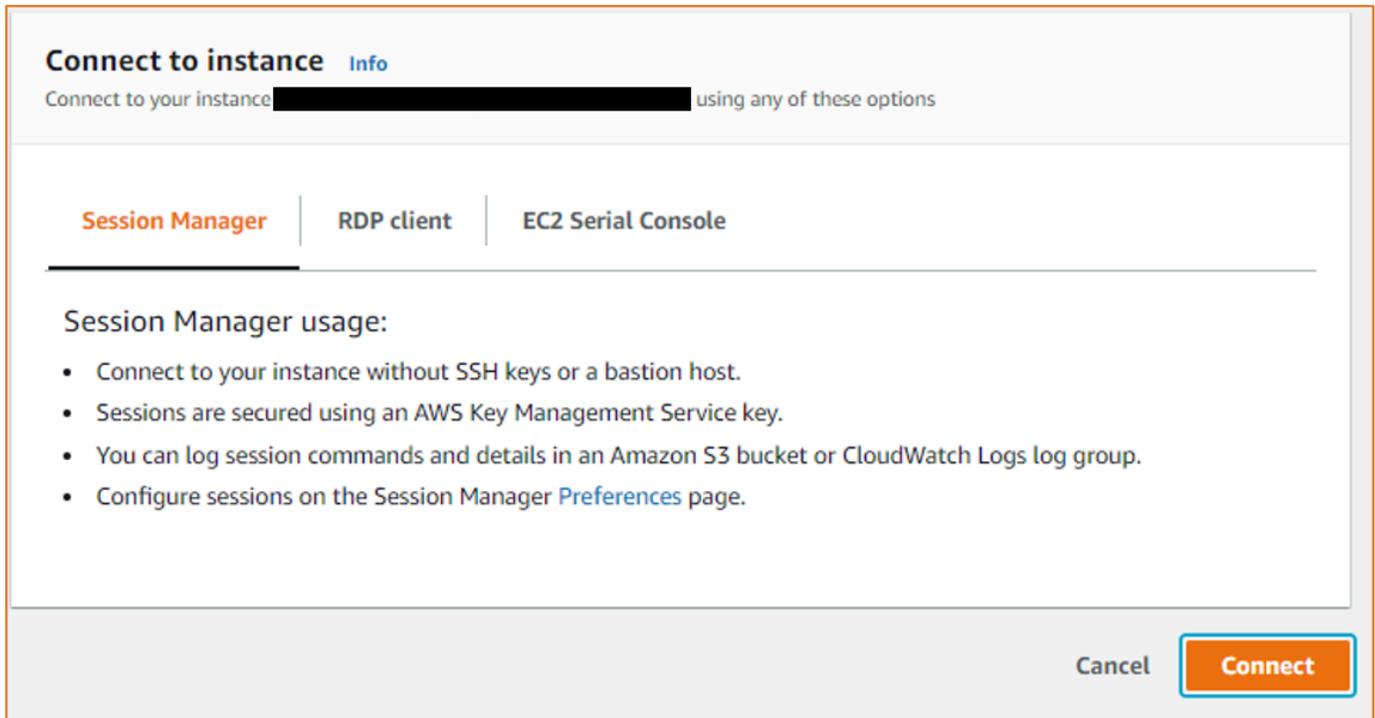
Session Manager adalah AWS Systems Manager kemampuan yang dikelola sepenuhnya untuk mengelola instans Amazon EC2 Anda melalui shell interaktif, satu-klik, berbasis browser, atau melalui AWS CLI. Anda dapat menggunakan Session Manager untuk memulai sesi dengan sebuah instans di akun Anda. Setelah sesi 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 .

Sebelum mencoba menghubungkan ke sebuah instans menggunakan Session Manager, pastikan bahwa langkah-langkah penyiapan yang diperlukan telah diselesaikan. Untuk informasi selengkapnya, lihat [Menyiapkan Session Manager](#).

Untuk menyambung ke instans Amazon EC2 menggunakan Session Manager 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 Session Manager.
5. Pilih Hubungkan.



i Tip

Jika Anda menerima kesalahan bahwa Anda tidak berhak untuk melakukan satu atau lebih tindakan Systems Manager (`ssm:command-name`), maka Anda harus memperbarui kebijakan agar dapat 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 .

Connect ke instans Anda menggunakan EC2 Instance Connect Endpoint

EC2 Instance Connect Endpoint memungkinkan Anda terhubung dengan aman ke instans dari internet, tanpa menggunakan host bastion, atau mengharuskan virtual private cloud (VPC) Anda memiliki konektivitas internet langsung.

Manfaat

- Anda dapat terhubung ke instans Anda tanpa mengharuskan instans memiliki alamat IPv4 publik. AWS mengenakan biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastis. 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 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 yang tidak berhasil, dicatat. [CloudTrail](#)

Harga

Tidak ada biaya tambahan untuk menggunakan EC2 Instance Connect Endpoints. Jika Anda menggunakan Titik Akhir Connect Instance EC2 untuk menyambung ke instans di Availability Zone yang berbeda, akan [dikenakan biaya tambahan untuk transfer data](#) di seluruh Availability Zone.

Daftar Isi

- [Cara kerjanya](#)
- [Pertimbangan](#)
- [Berikan izin untuk menggunakan EC2 Instance Connect Endpoint](#)
- [Grup keamanan untuk EC2 Instans Connect Endpoint](#)
- [Membuat EC2 Instance Connect Endpoint](#)
- [Connect ke instans Amazon EC2 menggunakan EC2 Instance Connect Endpoint](#)
- [Koneksi log dibuat melalui EC2 Instance Connect Endpoint](#)
- [Menghapus Titik Akhir Connect Instance EC2](#)
- [Peran tertaut layanan untuk EC2 Instance Connect Endpoint](#)
- [Kuota untuk EC2 Instance Connect Endpoint](#)

Cara kerjanya

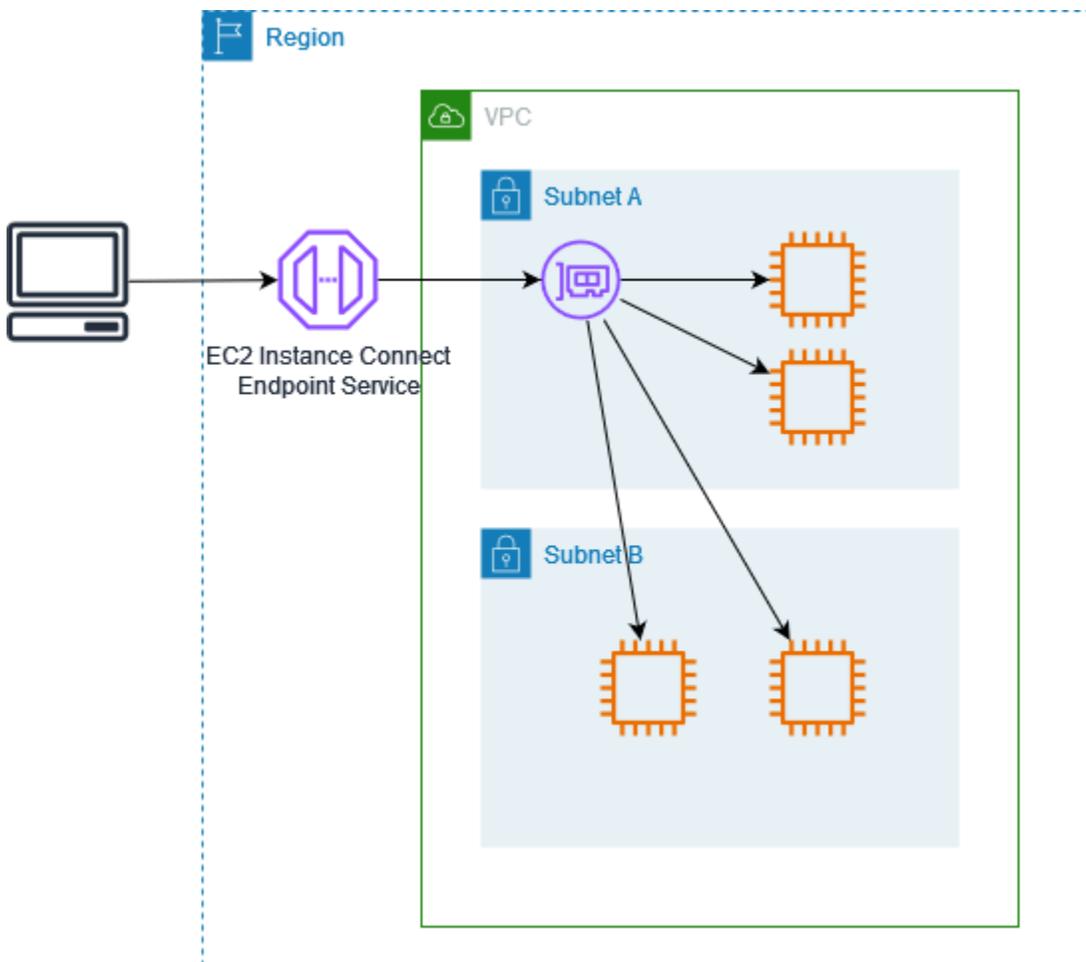
EC2 Instance Connect Endpoint adalah proxy TCP yang sadar identitas. Layanan Endpoint Connect Instance Connect EC2 membuat terowongan pribadi dari komputer Anda ke titik akhir menggunakan

kredensial 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 memungkinkan 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 endpoint di subnet, yang berfungsi sebagai titik masuk untuk lalu lintas yang ditujukan untuk 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

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.
- Instans Anda harus memiliki alamat IPv4 (baik privat maupun publik). EC2 Instance Connect Endpoint tidak mendukung koneksi ke instans menggunakan alamat IPv6.
- (Instance Linux) Jika Anda menggunakan key pair 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, lihat [Menginstal EC2 Instance Connect](#).
- Anda dapat menetapkan grup keamanan ke EC2 Instance Connect Endpoint saat membuatnya. Jika tidak, kami menggunakan grup keamanan default untuk VPC. Grup keamanan untuk Titik Akhir Connect Instans EC2 harus mengizinkan lalu lintas keluar ke instans tujuan. Untuk informasi selengkapnya, lihat [Grup keamanan untuk EC2 Instans Connect Endpoint](#).
- Anda dapat mengonfigurasi EC2 Instance Connect Endpoint 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 yang masuk.
 - Jika Anda mengaktifkan pelestarian IP klien, grup keamanan untuk instance harus mengizinkan lalu lintas dari klien. Selain itu, instans harus berada dalam VPC yang sama dengan EC2 Instance Connect Endpoint.
 - Jika Anda mematikan pelestarian IP klien, grup keamanan untuk instance harus mengizinkan lalu lintas dari VPC. Ini adalah opsi default.
 - Jenis contoh berikut tidak mendukung pelestarian IP klien: C1, CG1, CG2, G1, H1, M1, M2, M3, dan T1. Jika Anda mengaktifkan pelestarian IP klien dan mencoba menyambung ke instans dengan salah satu jenis instans ini menggunakan EC2 Instance Connect Endpoint, koneksi akan gagal.
 - Pelestarian IP klien tidak didukung saat lalu lintas dirutekan melalui gateway transit.
- Saat Anda membuat Titik Akhir Connect Instance EC2, peran yang ditautkan layanan akan dibuat secara otomatis untuk layanan Amazon EC2 di (IAM). AWS Identity and Access Management 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 [Peran tertaut layanan untuk EC2 Instance Connect Endpoint](#).
- Setiap EC2 Instance Connect Endpoint dapat mendukung hingga 20 koneksi bersamaan.

- Durasi maksimum untuk koneksi TCP yang ditetapkan adalah 1 jam (3.600 detik). Anda dapat menentukan durasi maksimum yang diizinkan dalam kebijakan IAM, yang dapat mencapai 3.600 detik. Untuk informasi selengkapnya, lihat [Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans](#).
- Titik Akhir Instance Connect EC2 tidak didukung di Canada West (Calgary).

Berikan izin untuk menggunakan EC2 Instance Connect Endpoint

Secara default, entitas IAM tidak memiliki izin untuk membuat, mendeskripsikan, atau memodifikasi Titik Akhir Connect Instance EC2. 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 bahwa Anda dapat mengontrol izin yang dimiliki pengguna untuk EC2 Instance Connect Endpoint.

Contoh

- [Izin untuk membuat, mendeskripsikan, dan menghapus EC2 Instance Connect Endpoint](#)
- [Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans](#)
- [Izin untuk terhubung hanya dari rentang alamat IP tertentu](#)

Izin untuk membuat, mendeskripsikan, dan menghapus EC2 Instance Connect Endpoint

Untuk membuat Titik Akhir Connect instans EC2, pengguna memerlukan izin untuk tindakan berikut:

- `ec2:CreateInstanceConnectEndpoint`
- `ec2:CreateNetworkInterface`
- `ec2:CreateTags`
- `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, mendeskripsikan, 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, bagian Resource memberikan izin untuk membuat 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.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "GrantAllActionsInAllSubnets",
    "Action": [
      "ec2:CreateInstanceConnectEndpoint",
      "ec2:DeleteInstanceConnectEndpoint",
      "ec2:CreateNetworkInterface",
      "ec2:CreateTags",
      "iam:CreateServiceLinkedRole"
    ],
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:region:account-id:subnet/*"
  },
  {
    "Action": [
      "ec2:CreateNetworkInterface"
    ],
    "Effect": "Allow",
    "Resource": "arn:aws:ec2::security-group/*"
  },
  {
    "Sid": "DescribeInstanceConnectEndpoints",
    "Action": [
```

```
        "ec2:DescribeInstanceConnectEndpoints"
    ],
    "Effect": "Allow",
    "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 ditetapkan. 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 [Connect ke instans Amazon EC2 menggunakan EC2 Instance Connect Endpoint](#).

Anda juga dapat memberikan akses pengguna untuk membuat koneksi ke instans berdasarkan keberadaan tag sumber daya pada Titik Akhir 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 sebuah 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 berikut kebijakan IAM mengizinkan prinsipal IAM untuk terhubung ke instance hanya menggunakan Titik Akhir 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 jika koneksi ke instance dibuat pada `—port 22 (SSH)`, jika alamat IP pribadi instance berada 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).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "EC2InstanceConnect",
    "Action": "ec2-instance-connect:OpenTunnel",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:region:account-id: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 jika koneksi ke instance dibuat pada port 3389 (RDP), jika alamat IP pribadi dari instance terletak dalam kisaran 10.0.1.0/31 (antara 10.0.1.0 dan 10.0.1.1), dan kurang dari atau sama maxTunnelDuration dengan detik. 3600 Koneksi terputus setelah 3600 detik (1 jam).

```

{
    "Version": "2012-10-17",
    "Statement": [{

```

```

    "Sid": "EC2InstanceConnect",
    "Action": "ec2-instance-connect:OpenTunnel",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:region:account-id: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 panggilan utama IAM `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.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "ec2-instance-connect:OpenTunnel",

```

```

    "Resource": "arn:aws:ec2:region:account-id: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 mencapai dan meninggalkan sumber daya yang terkait dengannya. Misalnya, kami menolak lalu lintas ke dan dari instans Amazon EC2 kecuali diizinkan secara khusus oleh grup keamanan yang terkait dengan instans tersebut.

Contoh berikut menunjukkan cara mengonfigurasi aturan grup keamanan untuk Titik Akhir Connect Instance Connect EC2 dan instance 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 Titik Akhir Connect Instans EC2 harus mengizinkan lalu lintas keluar yang ditujukan untuk instans target untuk meninggalkan titik akhir. Anda dapat menentukan grup keamanan instans atau rentang alamat IPv4 VPC sebagai tujuan.

Lalu lintas ke titik akhir berasal dari Layanan Titik Akhir Connect Instance Connect EC2, dan itu 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	<i>ID grup keamanan instans</i>	22	Mengizinkan lalu lintas SSH keluar ke semua instance yang terkait dengan grup keamanan instance

Contoh aturan keluar: rentang alamat IPv4

Contoh berikut memungkinkan lalu lintas keluar ke kisaran alamat IPv4 yang ditentukan. Alamat IPv4 dari sebuah instans ditetapkan dari subnetnya, sehingga Anda dapat menggunakan rentang alamat IPv4 dari VPC.

Protokol	Tujuan	Rentang port	Komentar
TCP	<i>VPC IPv4 CIDR</i>	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 Titik Akhir Connect Instans EC2. Anda dapat menentukan grup keamanan titik akhir atau rentang alamat IPv4 sebagai sumbernya. Jika Anda menentukan rentang alamat IPv4, sumbernya tergantung pada apakah pelestarian IP klien tidak aktif atau aktif. Untuk informasi selengkapnya, lihat [Pertimbangan](#).

Karena grup keamanan stateful, lalu lintas respons diizinkan untuk meninggalkan VPC terlepas dari aturan keluar untuk grup keamanan instance.

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, baik pelestarian IP klien aktif atau tidak aktif. 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	<i>ID grup keamanan endpoint</i>	22	Mengizinkan lalu lintas SSH masuk dari sumber daya yang terkait dengan grup keamanan titik akhir

Contoh aturan masuk: Pelestarian IP klien tidak aktif

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 endpoint ditetapkan dari subnetnya, sehingga Anda dapat menggunakan rentang alamat IPv4 dari VPC untuk memungkinkan koneksi ke semua instance di VPC.

Protokol	Sumber	Rentang port	Komentar
TCP	<i>VPC IPv4 CIDR</i>	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	<i>Rentang alamat IPv4 publik</i>	22	Mengizinkan lalu lintas masuk dari rentang alamat IPv4 klien yang ditentukan

Membuat EC2 Instance Connect Endpoint

Anda dapat membuat Instance Connect Endpoint EC2 untuk memungkinkan koneksi aman ke instans Anda.

Anda tidak dapat mengubah EC2 Instance Connect Endpoint setelah Anda membuatnya. Sebagai gantinya, Anda harus menghapus EC2 Instance Connect Endpoint dan membuat yang baru dengan pengaturan yang Anda butuhkan.

Prasyarat

Anda harus memiliki izin IAM yang diperlukan untuk membuat EC2 Instance Connect Endpoint. Untuk informasi selengkapnya, lihat [Izin untuk membuat, mendeskripsikan, dan menghapus EC2 Instance Connect Endpoint](#).

Subnet bersama

Anda dapat membuat Instance Connect Endpoint EC2 di subnet yang dibagikan dengan Anda. Anda tidak dapat menggunakan Instance Connect Endpoint EC2 yang dibuat oleh pemilik VPC di subnet yang dibagikan dengan Anda.

Buat titik akhir menggunakan konsol

Gunakan prosedur berikut untuk membuat EC2 Instance Connect Endpoint.

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 Kategori layanan, pilih EC2 Instance Connect Endpoint.
 - c. Untuk VPC, pilih VPC yang memiliki instance target.
 - d. (Opsional) Untuk mempertahankan alamat IP klien, perluas Pengaturan tambahan dan pilih kotak centang. Jika tidak, defaultnya adalah menggunakan antarmuka jaringan endpoint sebagai alamat IP klien.
 - 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. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
4. Tinjau pengaturan Anda dan kemudian pilih Buat titik akhir.

Status awal endpoint adalah Pending. Sebelum Anda dapat terhubung ke sebuah instance menggunakan endpoint ini, Anda harus menunggu hingga status endpoint tersedia. Hal ini dapat menghabiskan waktu beberapa menit.

5. Untuk terhubung ke instans menggunakan titik akhir Anda, lihat [Sambungkan ke instans](#).

Buat titik akhir menggunakan AWS CLI

Gunakan [create-instance-connect-endpoint](#) AWS CLI perintah.

Prasyarat

Instal AWS CLI versi 2 dan konfigurasi menggunakan kredensial Anda. Untuk informasi selengkapnya, lihat [Menginstal atau memperbarui ke versi terbaru AWS CLI](#) dan [Mengkonfigurasi AWS CLI](#) dalam Panduan AWS Command Line Interface Pengguna. Atau, buka AWS CloudShell dan jalankan AWS CLI perintah di shell yang telah diautentikasi sebelumnya.

Untuk membuat titik akhir

Gunakan perintah berikut untuk membuat antarmuka jaringan titik akhir untuk EC2 Instance Connect Endpoint di subnet yang ditentukan.

```
aws ec2 create-instance-connect-endpoint --subnet-id subnet-0123456789example
```

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",
  "CreatedAt": "2023-04-07T15:43:53.000Z",
  "SubnetId": "subnet-0123abcd",
  "PreserveClientIp": false,
  "SecurityGroupIds": [
    "sg-0123abcd"
  ],
  "Tags": []
}
```

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](#) AWS CLI perintah untuk memantau status EC2 Instance Connect Endpoint. Parameter `--query` memfilter 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
```

Connect 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.

Daftar Isi

- [Prasyarat](#)
- [Pemecahan Masalah](#)

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](#).
- Titik Akhir Instans Connect EC2 harus dalam status Tersedia (konsol) atau create-complete (AWS CLI). Jika Anda tidak memiliki EC2 Instance Connect Endpoint untuk VPC Anda, Anda dapat membuatnya. Untuk informasi selengkapnya, lihat [Membuat EC2 Instance Connect Endpoint](#).
- (Instans Linux) Untuk menggunakan konsol EC2 untuk terhubung ke instans Anda, atau menggunakan CLI untuk menghubungkan dan meminta EC2 Instance Connect menangani kunci sementara, instans Anda harus menginstal EC2 Instance Connect. Untuk informasi selengkapnya, lihat [Menginstal EC2 Instance Connect](#).
- Pastikan bahwa grup keamanan instans mengizinkan lalu lintas SSH masuk dari Titik Akhir Connect Instance Connect EC2. Untuk informasi selengkapnya, lihat [Aturan grup keamanan instans target](#).

Hubungkan ke instans Linux Anda menggunakan konsol Amazon EC2

Anda dapat terhubung ke instans menggunakan konsol Amazon EC2 sebagai berikut.

Untuk terhubung ke instans Anda menggunakan klien berbasis browser

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, pilih Connect.
4. Pilih tab EC2 Instance Connect.
5. Untuk Tipe koneksi, pilih Hubungkan menggunakan EC2 Instance Connect Endpoint.

6. Untuk EC2 Instance Connect Endpoint, pilih ID dari EC2 Instance Connect Endpoint.
7. Untuk Nama Pengguna, jika AMI yang Anda gunakan untuk meluncurkan instance menggunakan nama pengguna selain `ec2-user`, masukkan nama pengguna yang benar.
8. 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.

9. Pilih **Hubungkan**. Ini membuka jendela terminal untuk instance 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.

Untuk informasi tentang menggunakan AWS CLI to connect ke instans Anda menggunakan SSH, lihat [Connect 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-0123456789example \  
-o ProxyCommand='aws ec2-instance-connect open-tunnel --instance-  
id i-0123456789example'
```

Untuk:

- `-i` – Menentukan pasangan kunci yang digunakan untuk meluncurkan instans.
- `ec2-user@i-0123456789example` – Menentukan nama pengguna AMI yang digunakan untuk meluncurkan instans dan ID instans.

- `--instance-id` – Menentukan ID instans yang akan dihubungkan. Atau, tentukan `%h`, yang mengekstrak ID instans dari pengguna.

Multi-koneksi

Untuk mengizinkan beberapa koneksi ke sebuah instance, pertama-tama jalankan [open-tunnel](#) AWS CLI perintah untuk mulai mendengarkan koneksi TCP baru, dan kemudian gunakan `ssh` untuk membuat koneksi TCP baru dan terowongan pribadi ke instans Anda.

Untuk memungkinkan beberapa koneksi ke instans 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-0123456789example \  
  --local-port 8888
```

Output 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.
```

Connect ke instance Linux Anda menggunakan AWS CLI

Jika Anda hanya mengetahui ID instans Anda, Anda dapat menggunakan AWS CLI perintah [ec2-instance-connect untuk menyambung](#) ke instans Anda menggunakan klien SSH. Untuk informasi selengkapnya tentang menggunakan perintah [ec2-instance-connect](#), lihat [Connect menggunakan AWS CLI](#)

Prasyarat

Instal AWS CLI versi 2 dan konfigurasi menggunakan kredensial Anda. Untuk informasi selengkapnya, lihat [Menginstal atau memperbarui ke versi terbaru AWS CLI](#) dan [Mengkonfigurasi AWS CLI](#) dalam Panduan AWS Command Line Interface Pengguna. Atau, buka AWS CloudShell dan jalankan AWS CLI perintah di shell yang telah diautentikasi sebelumnya.

Untuk menghubungkan ke instans menggunakan ID instans dan EC2 Instance Connect Endpoint

Jika Anda hanya mengetahui ID instance, gunakan perintah CLI [ec2-instance-connect](#), dan tentukan ssh perintah, ID instance, dan parameter dengan nilainya. `--connection-type eice`

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

Tip

Jika Anda mendapatkan kesalahan saat menggunakan perintah ini, pastikan Anda menggunakan AWS CLI versi 2. `sshParameter` ini hanya tersedia di AWS CLI versi 2. Untuk informasi selengkapnya, lihat [Tentang AWS CLI versi 2](#) di Panduan AWS Command Line Interface Pengguna.

Connect ke instans Windows Anda menggunakan EC2 Instance Connect Endpoint

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 [Connect to instance Windows Anda menggunakan RDP](#). Setelah mengunduh file desktop RDP pada Langkah 8, Anda akan mendapatkan pesan Unable to connect, yang diharapkan karena instans 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-0123456789example \  
  --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. Klik kanan file desktop RDP dan pilih Edit.
5. Di jendela Edit PC, untuk nama PC (instance untuk terhubung), masukkan `localhost:local-port`, di mana `local-port` menggunakan nilai yang sama seperti yang Anda tentukan di Langkah 2, lalu pilih Simpan.

Perhatikan bahwa tangkapan layar berikut dari jendela Edit PC berasal dari Microsoft Remote Desktop di Mac. Jika Anda menggunakan klien Windows, jendelanya mungkin berbeda.

Edit PC

PC name: localhost:5555

User account: Administrator

General Display Devices & Audio Folders

Friendly name: windows-test

Group: Saved PCs

Gateway: No gateway

Bypass for local addresses

Reconnect if the connection is dropped

Connect to an admin session

Swap mouse buttons

Cancel Save

6. Di klien RDP, klik kanan PC (yang baru saja Anda konfigurasi) dan pilih Connect untuk terhubung ke instans 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 tips pemecahan masalah lainnya untuk instance Linux, lihat. [Memecahkan masalah saat menghubungkan ke instans Linux](#) Untuk tips pemecahan masalah untuk instance Windows, lihat. [the section called “Hubungkan ke instans Windows Anda”](#)

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 Titik Akhir Connect Instans EC2 dengan AWS CloudTrail log.

Untuk informasi selengkapnya tentang penggunaan AWS CloudTrail dengan Amazon EC2, lihat. [Log panggilan API Amazon EC2 menggunakan AWS CloudTrail](#)

Log panggilan EC2 Instance Connect Endpoint API dengan AWS CloudTrail

Operasi sumber daya Endpoint Instance Connect EC2 dicatat CloudTrail sebagai peristiwa manajemen. Saat panggilan API berikut dibuat, aktivitas akan direkam sebagai CloudTrail peristiwa dalam riwayat Acara:

- `CreateInstanceConnectEndpoint`
- `DescribeInstanceConnectEndpoints`
- `DeleteInstanceConnectEndpoint`

Anda dapat melihat, mencari, dan mengunduh acara terbaru di situs Anda Akun AWS. Untuk informasi selengkapnya, lihat [Melihat CloudTrail peristiwa dengan riwayat peristiwa](#) di Panduan AWS CloudTrail Pengguna.

Gunakan AWS CloudTrail untuk mengaudit pengguna yang terhubung ke EC2 Instance Connect Endpoint

Upaya koneksi ke instans melalui EC2 Instance Connect Endpoint dicatat CloudTrail dalam riwayat peristiwa. Ketika koneksi ke instans dimulai melalui Titik Akhir Connect Instance EC2, koneksi dicatat sebagai peristiwa CloudTrail manajemen dengan `eventName OpenTunnel`

Anda dapat membuat EventBridge aturan Amazon yang merutekan CloudTrail acara ke target. Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#).

Berikut ini adalah contoh peristiwa `OpenTunnel` manajemen yang masuk CloudTrail.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGHIJGONGNOM00CB6XYTQEXAMPLE",
    "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
    "accountId": "123456789012",
    "accessKeyId": "ABCDEFGHIJKZHN40SN2AEXAMPLE",
    "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"
}

```

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, dan menghapus EC2 Instance Connect Endpoint](#).

Saat Anda menghapus Instance Connect Endpoint EC2 menggunakan konsol, maka akan masuk ke status Menghapus. Jika penghapusan berhasil, titik akhir yang dihapus tidak lagi muncul. Jika penghapusan gagal, status **delete-failed** dan pesan Status memberikan alasan kegagalan.

Saat Anda menghapus Instance Connect Endpoint EC2 menggunakan AWS CLI, titik tersebut akan memasuki status. `delete-in-progress` Jika penghapusan berhasil, ia memasuki negara bagian. `delete-complete` Jika penghapusan gagal, negara 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-endpoints](#) AWS CLI perintah dan tentukan ID dari EC2 Instance Connect Endpoint untuk dihapus.

```
aws ec2 delete-instance-connect-endpoint --instance-connect-endpoint-id eice-03f5e49b83924bbc7
```

Contoh Output

```
{
  "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",
    "CreatedAt": "2023-02-07T12:05:37+00:00",
    "SubnetId": "subnet-0123abcd"
  }
}
```

Peran tertaut layanan untuk EC2 Instance Connect Endpoint

[Amazon EC2 menggunakan peran terkait layanan AWS Identity and Access Management \(IAM\).](#)

Peran tertaut layanan adalah tipe peran IAM unik yang tertaut langsung ke Amazon EC2. Peran terkait layanan telah ditentukan sebelumnya oleh Amazon EC2 dan menyertakan semua izin yang diperlukan agar Amazon EC2 dapat memanggil orang lain atas nama Anda. Layanan AWS Untuk informasi selengkapnya, lihat [Menggunakan peran terkait layanan](#) dalam Panduan Pengguna IAM.

Izin peran terkait layanan 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 `AWSServiceRoleForEC2InstanceConnect` terkait layanan mempercayai layanan berikut untuk mengambil peran:

- `ec2-instance-connect.amazonaws.com`

Peran `AWSServiceRoleForEC2InstanceConnect` terkait layanan menggunakan Endpoint `InstanceConnectEc2` kebijakan terkelola. Untuk melihat izin kebijakan ini, lihat [InstanceConnectTitik Akhir Ec2](#) di Referensi Kebijakan AWS Terkelola.

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat [Izin peran tertaut layanan](#) dalam Panduan Pengguna IAM.

Membuat peran terkait layanan untuk EC2 Instance Connect Endpoint

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda membuat Titik Akhir Connect Instance EC2, Amazon EC2 akan membuat peran terkait layanan untuk Anda.

Mengedit peran terkait layanan untuk EC2 Instance Connect Endpoint

Titik Akhir Instance Connect EC2 tidak memungkinkan 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 yang ditautkan `AWSServiceRoleForEC2InstanceConnect` layanan.

Anda harus menghapus semua sumber daya Titik Akhir Instance Connect EC2 sebelum dapat menghapus peran terkait layanan.

Untuk menghapus peran terkait layanan, lihat [Menghapus peran terkait layanan](#) di Panduan Pengguna IAM.

Kuota untuk EC2 Instance Connect Endpoint

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk setiap layanan. AWS Kecuali dinyatakan lain, setiap kuota bersifat khusus per Wilayah.

Anda Akun AWS memiliki kuota berikut yang terkait dengan EC2 Instance Connect Endpoint.

Deskripsi	Kuota
Jumlah maksimum EC2 Instance Connect Endpoint per per Akun AWS Wilayah AWS	5
Jumlah Maksimum EC2 instans Connect per VPC	1
Jumlah maksimum EC2 instans Connect Endpoint per subnet	1
Jumlah maksimum koneksi bersamaan per EC2 Instance Connect Endpoint	20

Hubungkan instans EC2 Anda ke sumber daya AWS

Setelah meluncurkan instans, Anda dapat menghubungkannya ke satu atau lebih AWS sumber daya.

Bagian ini menjelaskan cara menghubungkan instans Amazon EC2 secara otomatis ke basis data Amazon RDS.

Hubungkan instans EC2 ke basis data RDS secara otomatis

Gunakan fitur koneksi otomatis di konsol Amazon EC2 untuk menghubungkan satu atau lebih instans EC2 dengan cepat ke basis data RDS untuk memungkinkan lalu lintas di antaranya.

Untuk informasi selengkapnya, lihat [Bagaimana koneksi dikonfigurasi secara otomatis](#). Untuk panduan detail, yang mencakup cara lain untuk menghubungkan instans EC2 dan basis data RDS, lihat [Tutorial: Hubungkan instans Amazon EC2 ke basis data Amazon RDS](#).

Topik

- [Biaya](#)
- [Prasyarat](#)
- [Hubungkan instans dan basis data secara otomatis](#)
- [Bagaimana koneksi dikonfigurasi secara otomatis](#)

Biaya

Meskipun tidak ada biaya untuk secara otomatis menghubungkan instans EC2 Anda ke basis data RDS, Anda dikenai biaya untuk layanan yang dasarnya. Biaya transfer data akan berlaku jika instans EC2 dan basis data RDS Anda berada di Zona Ketersediaan yang berbeda. Untuk informasi tentang biaya transfer data, lihat [Transfer Data](#) di halaman Harga Sesuai Permintaan Amazon EC2.

Prasyarat

Sebelum Anda dapat secara otomatis menghubungkan instans EC2 ke basis data RDS, periksa hal berikut:

- Instans EC2 harus dalam status Berjalan. Anda tidak dapat menghubungkan instans EC2 jika berada dalam status lain.
- Instans EC2 dan basis data RDS harus berada di cloud privat virtual (VPC) yang sama. Fitur koneksi otomatis tidak didukung jika instans EC2 dan basis data RDS berada di VPC yang berbeda.

Hubungkan instans dan basis data secara otomatis

Anda dapat secara otomatis menghubungkan instans EC2 ke basis data RDS segera setelah Anda meluncurkan instans Anda, atau nanti.

Hubungkan secara otomatis segera setelah peluncuran

Gunakan langkah-langkah berikut untuk secara otomatis menghubungkan instans EC2 ke basis data RDS segera setelah Anda meluncurkan instans EC2.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS](#).

Untuk secara otomatis menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS menggunakan konsol EC2

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari dasbor konsol, pilih Luncurkan instans, lalu ikuti langkah-langkah untuk [meluncurkan instans](#).
3. Pada halaman konfirmasi peluncuran instans, pilih Hubungkan dengan basis data RDS.
4. Pada kotak dialog Hubungkan Basis Data RDS, lakukan hal berikut:
 - a. Untuk Peran basis data, pilih Klaster atau Instans.
 - b. Untuk Basis data RDS, pilih basis data yang akan dihubungkan.

 Note

Instans EC2 dan basis data RDS harus dalam VPC yang sama agar dapat saling terhubung.

- c. Pilih Hubungkan.

Lihat animasi: Menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is divided into several sections:

- Resources:** A summary table of EC2 resources in the Europe (Stockholm) Region. A table below lists: Instances (running) 1, Dedicated Hosts 0, Elastic IPs 0, Instances 1, Key pairs 1, Load balancers 0, Placement groups 0, Security groups 9, Snapshots 1, and Volumes 2.
- Launch instance:** A section with a 'Launch instance' button and a 'Migrate a server' link. A note states: 'Your instances will launch in the Europe (Stockholm) Region'.
- Scheduled events:** A section showing 'Europe (Stockholm)' with 'No scheduled events'.
- Migrate a server:** A section with the text: 'Use AWS Application Migration Service to simplify and expedite migration'.
- Service health:** Shows 'Region: Europe (Stockholm)' and 'Status: This service is operating normally'.
- Zones:** A table listing available zones in the region.

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

Hubungkan instans yang ada secara otomatis

Gunakan langkah-langkah berikut untuk secara otomatis menghubungkan instans EC2 yang ada ke basis data RDS.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Secara otomatis menghubungkan instans EC2 yang ada ke basis data RDS](#).

Untuk secara otomatis menghubungkan instans EC2 yang ada ke basis data RDS menggunakan konsol EC2

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih satu atau beberapa instans EC2 untuk terhubung ke basis data RDS, lalu pilih Tindakan, Jaringan, Hubungkan basis data RDS.

Jika basis data Connect RDS tidak tersedia, periksa apakah instans EC2 berada dalam status Berjalan dan berada dalam VPC yang sama.

4. Pada kotak dialog Hubungkan Basis Data RDS, lakukan hal berikut:
 - a. Untuk Peran basis data, pilih Klaster atau Instans.
 - b. Untuk Basis data RDS, pilih basis data yang akan dihubungkan.

Note

Instans EC2 dan basis data RDS harus dalam VPC yang sama agar dapat saling terhubung.

- c. Pilih Hubungkan.

Lihat animasi: Secara otomatis menghubungkan instans EC2 yang ada ke basis data RDS

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is divided into several panels:

- Resources:** A table showing EC2 resources in the Europe (Stockholm) region.

Resource	Count
Instances (running)	2
Instances	2
Placement groups	0
Volumes	3
Dedicated Hosts	0
Key pairs	1
Security groups	10
Elastic IPs	0
Load balancers	0
Snapshots	1
- Launch instance:** A section with a 'Launch Instance' button and a 'Migrate a server' button. It includes a note: 'Your instances will launch in the Europe (Stockholm) Region'.
- Scheduled events:** A section showing 'Europe (Stockholm)' with 'No scheduled events'.
- Migrate a server:** A section with the text: 'Use AWS Application Migration Service to simplify and expedite migration'.
- Service health:** A section showing 'Region: Europe (Stockholm)' and 'Status: This service is operating normally'.
- Zones:** A table listing available zones.

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3
- Account attributes:** A section on the right showing 'Supported platforms' (VPC), 'Default VPC' (vpc-78678c11), and various settings like EBS encryption, Zones, and EC2 Serial Console.
- Explore AWS:** A section on the right with promotional cards for 'Amazon GuardDuty Malware Protection', 'Enable Best Price-Performance with AWS Graviton2', and 'Get Up to 40% Better Price Performance'.
- Additional information:** A section at the bottom right with a 'Getting started guide' link.

Untuk informasi tentang cara menggunakan konsol Amazon RDS untuk menghubungkan instans EC2 secara otomatis ke basis data RDS, lihat [Mengonfigurasi konektivitas jaringan otomatis dengan instans EC2](#) di Panduan Pengguna Amazon RDS.

Bagaimana koneksi dikonfigurasi secara otomatis

Saat Anda menggunakan konsol EC2 untuk secara otomatis mengonfigurasi koneksi antara instans EC2 dan basis data RDS untuk memungkinkan lalu lintas di antara mereka, koneksi dikonfigurasi oleh [grup keamanan](#).

Grup keamanan secara otomatis dibuat dan ditambahkan ke instans EC2 dan basis data RDS, sebagai berikut:

- Amazon EC2 membuat grup keamanan yang disebut `ec2-rds-x` dan menambahkannya ke instans EC2. Ini memiliki satu aturan keluar yang memungkinkan lalu lintas ke basis data dengan menentukan `rds-ec2-x` (grup keamanan basis data) sebagai tujuannya.
- Amazon RDS membuat grup keamanan yang disebut `rds-ec2x` dan menambahkannya ke basis data. Grup keamanan ini memiliki satu aturan masuk yang memungkinkan lalu lintas dari instans EC2 dengan menentukan `ec2-rds-x` (grup keamanan instans EC2) sebagai sumbernya.

Kelompok keamanan merujuk satu sama lain sebagai tujuan dan sumber, dan hanya mengizinkan lalu lintas pada port basis data. Anda dapat menggunakan kembali grup keamanan ini sehingga basis data apa pun dengan grup keamanan `rds-ec2-x` dapat berbicara dengan instans EC2 apa pun dengan grup keamanan `ec2-rds-x`.

Nama grup keamanan mengikuti pola. Untuk grup keamanan yang dibuat oleh Amazon EC2, polanya adalah `ec2-rds-x`, dan untuk grup keamanan yang dibuat oleh Amazon RDS, polanya adalah `rds-ec2-x`. **x** adalah angka, yang bertambah 1 setiap kali grup keamanan baru dibuat secara otomatis.

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 AWS Management Console.

Ada berbagai opsi untuk mengonfigurasi koneksi. Dalam tutorial ini, kami mengeksplorasi tiga opsi berikut ini:

- [Opsi 2: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda menggunakan konsol ECA](#)

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.

- [Ops 2: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda 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.

- [Ops 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan meniru fitur koneksi otomatis](#)

Konfigurasi 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.

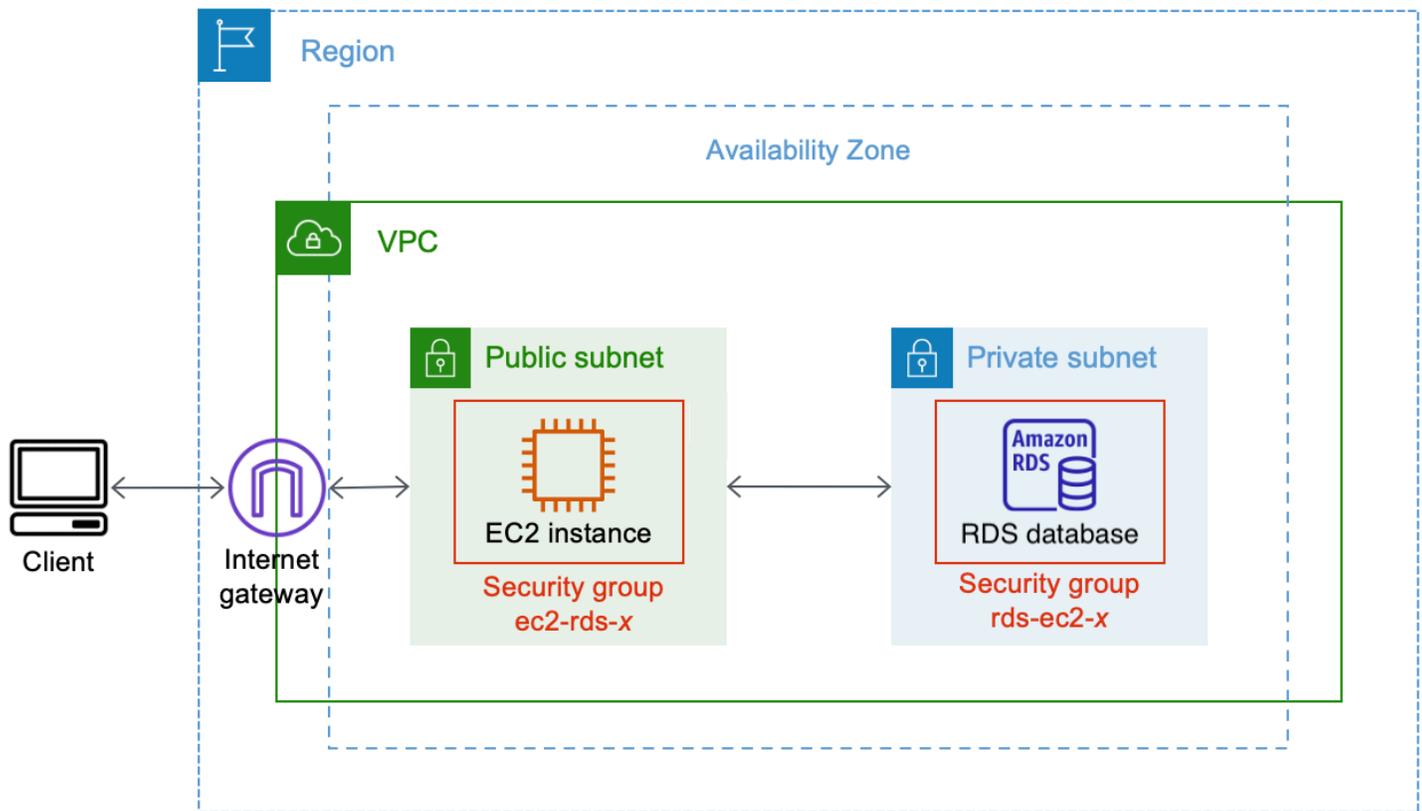


Diagram ini menggambarkan sumber daya berikut yang akan Anda buat:

- Anda akan membuat instans EC2 dan database RDS di Wilayah AWS VPC, 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 instans 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 instans EC2. Grup keamanan ini memiliki satu aturan keluar yang mereferensikan grup keamanan `rds-ec2-x` sebagai tujuannya. Dengan demikian, lalu lintas dari instans EC2 dapat menjangkau basis data RDS dengan grup keamanan `rds-ec2-x`.
- Grup keamanan `ec2-rds-x` dibuat dan ditambahkan ke basis data RDS. Grup keamanan ini memiliki satu aturan ke dalam yang mereferensikan grup keamanan `ec2-rds-x` sebagai sumbernya. Dengan demikian, lalu lintas dari instans EC2 dengan grup keamanan `ec2-rds-x` dapat menjangkau basis data 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

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 terletak di Wilayah yang sama.
- Dua grup keamanan – Konektivitas antara instans EC2 dan basis data RDS dikonfigurasi 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 dikenakan biaya untuk AWS sumber daya yang Anda buat.

Anda dapat menggunakan Amazon EC2 di bawah [tingkat gratis](#) asalkan AWS akun Anda berusia kurang dari 12 bulan dan Anda mengonfigurasi sumber daya sesuai dengan persyaratan tingkat gratis.

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 Harga Sesuai Permintaan 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 [Bersihkan](#).

Opsi 2: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda menggunakan konsol ECA

Tujuan

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.

Sebelum Anda memulai

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 untuk menyelesaikan Opsi 1

- [Tugas 1: Buat basis data RDS — opsional](#)
- [Tugas 2: Luncurkan instans EC2 — opsional](#)
- [Tugas 3: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda](#)
- [Tugas 4: Verifikasi konfigurasi koneksi](#)

Tugas 1: Buat basis data RDS — opsional

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.

Tujuan tugas

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. Jika Anda memiliki basis data RDS yang dapat digunakan, Anda dapat melewati tugas ini.

Important

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.

Langkah-langkah untuk membuat basis data RDS

Gunakan langkah-langkah berikut untuk membuat basis data RDS.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Membuat basis data RDS](#).

Konfigurasi basis data RDS

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 basis data 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 tanpa biaya. Namun, perhatikan bahwa tingkat gratis hanya tersedia jika akun Anda berusia kurang dari 12 bulan. Pembatasan lain berlaku. 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. Pada Konfigurasi instans, untuk Kelas instans DB, biarkan default, yaitu `db.t3.micro`. Jika akun Anda kurang dari 12 bulan, Anda dapat menggunakan kelas basis data ini secara gratis. Pembatasan lain berlaku. 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

The screenshot shows the Amazon RDS console interface. On the left is a navigation sidebar with the following items: Dashboard (highlighted), Databases, Performance insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, Event subscriptions, and Certificate update. The main content area features a top banner with an information icon and text: "Try the new Amazon RDS Multi-AZ deployment option for MySQL and PostgreSQL. For your Amazon RDS for MySQL and PostgreSQL workloads, improve transactional commit latencies instances by deploying the Multi-AZ DB cluster. [Learn more](#)". Below this is a prominent orange "Create database" button with a mouse cursor hovering over it, and a link that says "Or, Restore Multi-AZ DB Cluster from Snapshot".

Below the banner is a "Resources" section titled "Resources" with the subtext "You are using the following Amazon RDS resources in the EU (Stockholm) region (used/quota)". It lists various resource categories and their usage:

- DB Instances (3/40)**: Allocated storage (0.3 TB/100 TB), [Increase DB Instances limit](#)
- DB Clusters (1/40)**
- Reserved instances (0/40)**
- Snapshots (1)**
 - Manual**
 - DB Cluster (0/100)
 - DB Instance (0/100)
 - Automated**
 - DB Cluster (1)
 - DB Instance (0)
- Recent events (5)**
- Event subscriptions (0/20)**
- Parameter groups (2)**: Default (2), Custom (0/100)
- Option groups (1)**: Default (1), Custom (0/20)
- Subnet groups (1/50)**
- Supported platforms VPC**
- Default network vpc-78678c**

At the bottom of the main content area is a "Create database" section with the introductory text: "Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database i".

Anda sekarang siap untuk [Tugas 2: Luncurkan instans EC2 — opsional](#).

Tugas 2: Luncurkan instans EC2 — opsional

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 tugas

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. Jika Anda memiliki instans EC2 yang dapat digunakan, Anda dapat melewati tugas ini.

Important

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.

Langkah-langkah untuk meluncurkan instans EC2

Gunakan langkah-langkah berikut untuk meluncurkan instans EC2 untuk tutorial ini.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Luncurkan instans EC2](#).

Konfigurasi instans EC2

Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **tutorial-instance-1**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Penetapan otomatis IP publik: Aktif
- 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.

Untuk 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 basis data 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

Anda dapat menggunakan Amazon EC2 di bawah [tingkat Gratis](#) asalkan AWS akun Anda berusia kurang dari 12 bulan dan Anda memilih jenis `t2.micro` instans (atau `t3.micro` di Wilayah yang tidak `t2.micro` tersedia).

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 bahwa instans Anda menerima alamat IP publik, untuk Tetapkan otomatis IP publik, periksa apakah Aktifkan dipilih. Jika Nonaktifkan dipilih, pilih Edit

(di sebelah kanan Pengaturan Jaringan), lalu untuk Tetapkan otomatis IP 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 instans Anda, pada Firewall (grup keamanan), dari daftar tarik turun di samping kotak centang Izinkan 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 [Pemecahan masalah peluncuran instans](#).

Untuk informasi tentang peluncuran instans, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Lihat animasi: Luncurkan instans EC2

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is divided into several sections:

- Resources:** A summary of EC2 resources in the Europe (Stockholm) Region. It includes a table with the following data:

Resource	Count
Instances (running)	2
Dedicated Hosts	0
Elastic IPs	0
Instances	2
Key pairs	1
Load balancers	0
Placement groups	0
Security groups	10
Snapshots	1
Volumes	3
- Launch instance:** A section with a "Launch instance" button and a "Migrate a server" link. Below it is a note: "Note: Your instances will launch in the Europe (Stockholm) Region".
- Scheduled events:** A section showing "Europe (Stockholm)" with "No scheduled events".
- Service health:** A section showing the region "Europe (Stockholm)" with a status of "This service is operating normally". Below it is a table of zones:

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

Anda sekarang siap untuk [Tugas 3: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda](#).

Tugas 3: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda

Tujuan tugas

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.

Langkah-langkah untuk menghubungkan instans EC2 dan basis data RDS Anda

Gunakan langkah-langkah berikut untuk menghubungkan instans EC2 dan basis data RDS menggunakan fitur otomatis di konsol EC2.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS](#).

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.

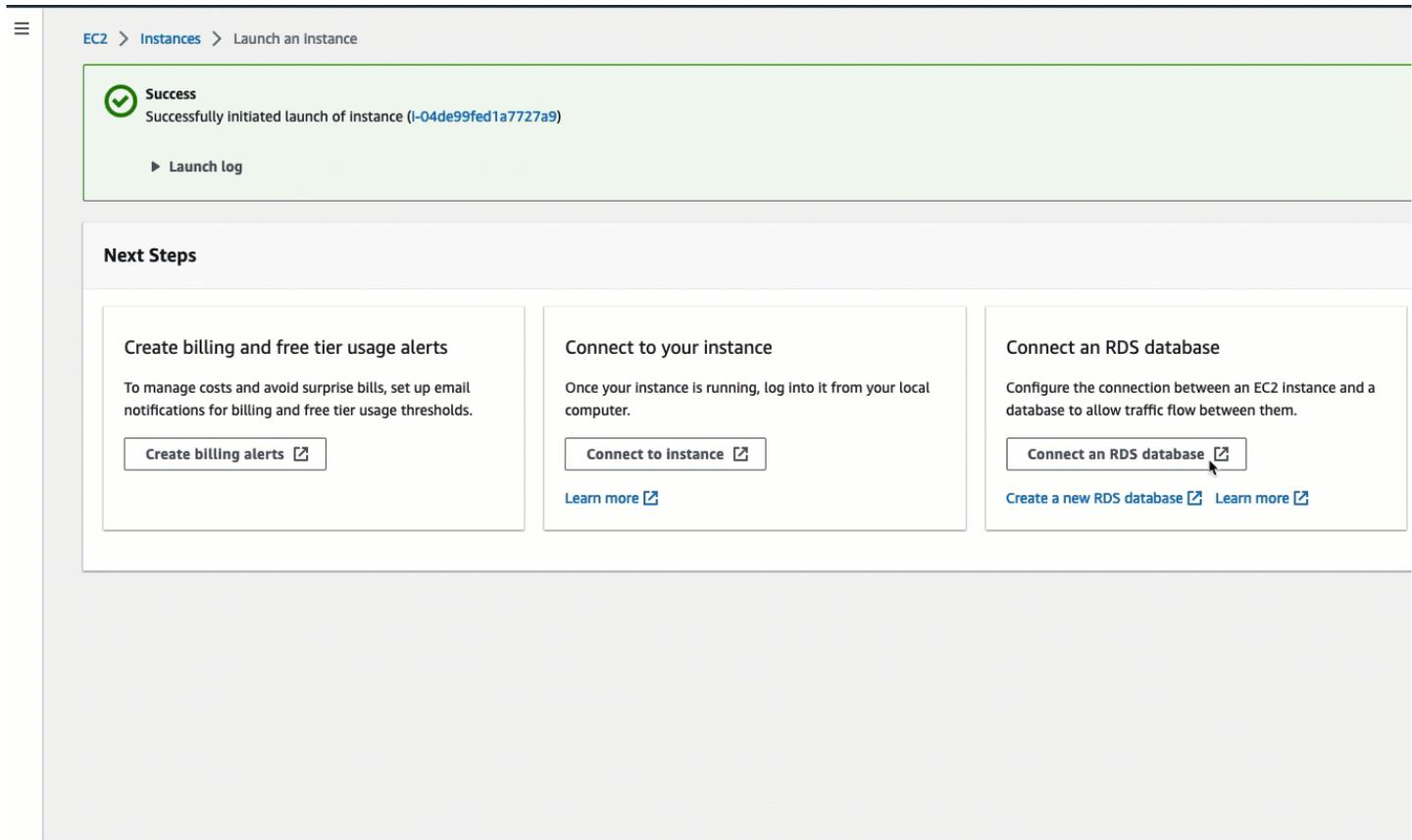
2. Untuk Peran basis data, pilih Instans. Instans dalam hal ini mengacu pada instans basis data.
3. Untuk basis data RDS, pilih basis data RDS yang Anda buat di Tugas 1.

 Note

Instans EC2 dan basis data RDS harus dalam VPC yang sama agar dapat saling terhubung.

4. Pilih Hubungkan.

Lihat animasi: Menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS



Anda sekarang siap untuk [Tugas 4: Verifikasi konfigurasi koneksi](#).

Tugas 4: Verifikasi konfigurasi koneksi

Tujuan tugas

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 EC2 untuk mengonfigurasi konektivitas, grup keamanan secara otomatis dibuat dan ditetapkan ke instans dan basis data, sebagai berikut:

- Grup keamanan `ec2-rds-x` dibuat dan ditambahkan ke basis data RDS. Grup keamanan ini memiliki satu aturan ke dalam yang mereferensikan grup keamanan `ec2-rds-x` sebagai sumbernya. Dengan demikian, lalu lintas dari instans EC2 dengan grup keamanan `ec2-rds-x` dapat menjangkau basis data RDS.
- Grup keamanan `ec2-rds-x` dibuat dan ditambahkan ke instans EC2. Grup keamanan ini memiliki satu aturan keluar yang mereferensikan grup keamanan `rds-ec2-x` sebagai tujuannya. Dengan

demikian, lalu lintas dari instans EC2 dapat menjangkau basis data RDS dengan grup keamanan `rds-ec2-x`.

Langkah-langkah untuk memverifikasi konfigurasi koneksi

Gunakan langkah-langkah berikut untuk memverifikasi konfigurasi koneksi.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Verifikasi konfigurasi koneksi](#).

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. Di tab Konektivitas & keamanan, pada Keamanan, Grup keamanan VPC, verifikasi bahwa grup keamanan yang disebut `rds-ec2-x` ditampilkan.
5. Pilih grup keamanan `rds-ec2-x`. Layar Grup Keamanan di konsol EC2 terbuka.
6. Pilih grup keamanan `rds-ec2-x` 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-langkah sebelumnya.
 - Deskripsi: Aturan untuk mengizinkan koneksi dari instans EC2 dengan `sg-1234567890example` terlampir
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. Pada Detail keamanan, Grup keamanan, verifikasi bahwa grup keamanan yang disebut `ec2-rds-x` ada dalam daftar. `x` adalah angka.
13. Pilih grup keamanan `rds-ec2-x` 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

The screenshot displays the AWS Management Console Home page. At the top, there is a 'Console Home' header with an 'Info' link, a 'Reset to default layout' button, and an 'Add widgets' button. Below the header is a 'Recently visited' section containing a grid of service tiles: RDS, EC2, IAM, VPC, Support, AWS FIS, EFS, Key Management Service, S3, AWS Marketplace Subscriptions, CloudFormation, and Directory Service. A 'View all services' link is located at the bottom of this grid. Below the 'Recently visited' section are two widgets. The 'Welcome to AWS' widget features a rocket icon and two links: 'Getting started with AWS' and 'Training and certification'. The 'AWS Health' widget shows three categories of issues: 'Open issues' (0), 'Scheduled changes' (0), and 'Other notifications' (0), each with a time filter for 'Past 7 days' or 'Upcoming and past 7 days'.

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.

Opsi 2: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda menggunakan konsol RDS

Tujuan

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.

Sebelum Anda memulai

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 untuk menyelesaikan Opsi 2

- [Tugas 1: Luncurkan instans EC2 – opsional](#)
- [Tugas 2: Buat basis data RDS dan secara otomatis hubungkan ke instans EC2 Anda](#)
- [Tugas 3: Verifikasi konfigurasi koneksi](#)

Tugas 1: Luncurkan instans EC2 – opsional

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 tugas

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. Jika Anda memiliki instans EC2 yang dapat digunakan, Anda dapat melewati tugas ini.

Langkah-langkah untuk meluncurkan instans EC2

Gunakan langkah-langkah berikut untuk meluncurkan instans EC2 untuk tutorial ini.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Luncurkan instans EC2](#).

Konfigurasi instans EC2

Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **tutorial-instance-2**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Penetapan otomatis IP publik: Aktif
- 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.

Untuk 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`.

ℹ Note

Anda dapat menggunakan Amazon EC2 di bawah [tingkat Gratis](#) asalkan AWS akun Anda berusia kurang dari 12 bulan dan Anda memilih jenis `t2.micro` instans (atau `t3.micro` di Wilayah yang tidak `t2.micro` tersedia).

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 bahwa instans Anda menerima alamat IP publik, untuk Tetapkan otomatis IP publik, periksa apakah Aktifkan dipilih. Jika Nonaktifkan dipilih, pilih Edit (di sebelah kanan Pengaturan Jaringan), lalu untuk Tetapkan otomatis IP 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 instans Anda, pada Firewall (grup keamanan), dari daftar tarik turun di samping kotak centang Izinkan 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 [Pemecahan masalah peluncuran instans](#).

Untuk informasi tentang peluncuran instans, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Lihat animasi: Luncurkan instans EC2

The screenshot shows the AWS Management Console interface for EC2 resources in the Europe (Stockholm) Region. The left sidebar contains navigation options like 'EC2 Dashboard', 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The main content area is divided into several sections:

- Resources:** A summary table showing the following counts:

Instances (running)	2	Dedicated Hosts	0	Elastic IPs	0
Instances	2	Key pairs	1	Load balancers	0
Placement groups	0	Security groups	10	Snapshots	1
Volumes	3				
- Launch instance:** A section with a 'Launch instance' button and a 'Migrate a server' link. Below it, a note states: 'Note: Your instances will launch in the Europe (Stockholm) Region'.
- Scheduled events:** A section showing 'Europe (Stockholm)' with 'No scheduled events'.
- Service health:** A section showing the region 'Europe (Stockholm)' with a status of 'This service is operating normally'.
- Zones:** A table listing available zones:

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

Anda sekarang siap untuk [Tugas 2: Buat basis data RDS dan secara otomatis hubungkan ke instans EC2 Anda](#).

Tugas 2: Buat basis data RDS dan secara otomatis hubungkan ke instans EC2 Anda

Tujuan tugas

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 untuk membuat basis data RDS

Gunakan langkah-langkah berikut untuk membuat basis data RDS dan menghubungkannya ke instans EC2 Anda menggunakan fitur otomatis di konsol RDS.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Membuat basis data RDS dan secara otomatis menghubungkannya ke instans EC2..](#)

Konfigurasi instans DB

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 Region (di kanan atas), pilih Wilayah AWS tempat Anda membuat instans 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 berusia kurang dari 12 bulan. Pembatasan lain berlaku. 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. Pada Konfigurasi instans, untuk Kelas instans DB, biarkan default, yaitu db.t3.micro. Jika akun Anda kurang dari 12 bulan, Anda dapat menggunakan instans ini gratis. Pembatasan lain berlaku. 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.

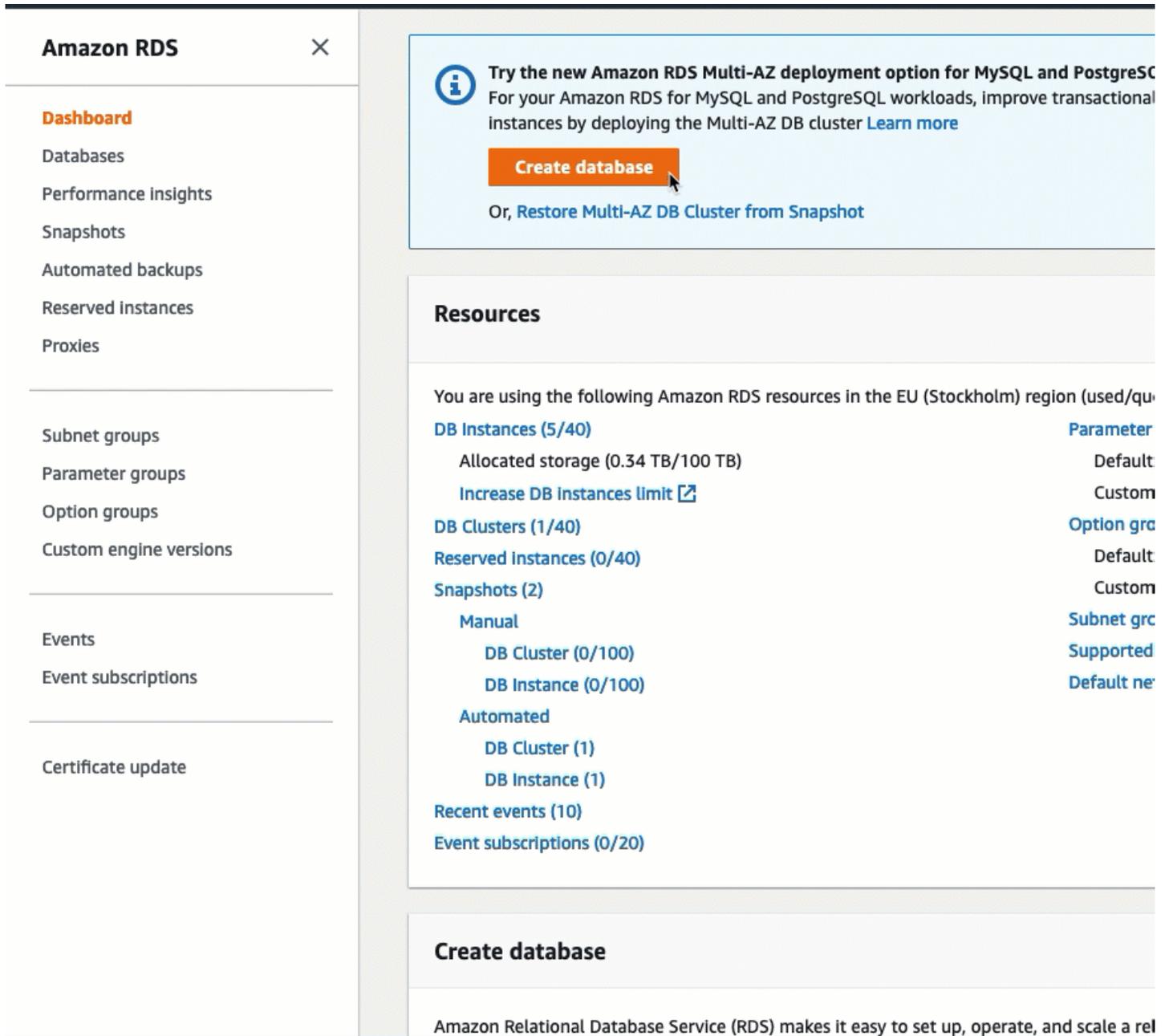
Saat Anda menggunakan fitur koneksi otomatis, grup keamanan ditambahkan ke instans EC2 ini, dan grup keamanan lain ditambahkan ke basis data RDS. Grup keamanan secara otomatis dikonfigurasi untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS. Pada tugas berikutnya, Anda akan memverifikasi 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.



The screenshot shows the Amazon RDS console interface. On the left is a navigation sidebar with the following items: Dashboard (highlighted), Databases, Performance Insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, Event subscriptions, and Certificate update. The main content area features a top banner with an information icon and text: "Try the new Amazon RDS Multi-AZ deployment option for MySQL and PostgreSQL. For your Amazon RDS for MySQL and PostgreSQL workloads, improve transactional instances by deploying the Multi-AZ DB cluster. Learn more". Below this is a prominent orange "Create database" button with a mouse cursor hovering over it, and a link "Or, Restore Multi-AZ DB Cluster from Snapshot". The "Resources" section lists usage in the EU (Stockholm) region: DB Instances (5/40), Allocated storage (0.34 TB/100 TB), DB Clusters (1/40), Reserved instances (0/40), and Snapshots (2). The snapshots are categorized into Manual (DB Cluster 0/100, DB Instance 0/100) and Automated (DB Cluster 1, DB Instance 1). At the bottom, there is a "Create database" section with the text: "Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a rel".

Anda sekarang siap untuk [Tugas 3: Verifikasi konfigurasi koneksi](#).

Tugas 3: Verifikasi konfigurasi koneksi

Tujuan tugas

Tujuan dari tugas ini adalah untuk memverifikasi bahwa dua grup keamanan dibuat dan ditetapkan ke instans serta basis data.

Saat Anda menggunakan fitur koneksi otomatis di konsol RDS untuk mengonfigurasi konektivitas, grup keamanan secara otomatis dibuat dan ditetapkan ke instans dan basis data, sebagai berikut:

- Grup keamanan `ec2-rds-x` dibuat dan ditambahkan ke basis data RDS. Grup keamanan ini memiliki satu aturan ke dalam yang mereferensikan grup keamanan `ec2-rds-x` sebagai sumbernya. Dengan demikian, lalu lintas dari instans EC2 dengan grup keamanan `ec2-rds-x` dapat menjangkau basis data RDS.
- Grup keamanan `ec2-rds-x` dibuat dan ditambahkan ke instans EC2. Grup keamanan ini memiliki satu aturan keluar yang mereferensikan grup keamanan `rds-ec2-x` sebagai tujuannya. Dengan demikian, lalu lintas dari instans EC2 dapat menjangkau basis data RDS dengan grup keamanan `rds-ec2-x`.

Langkah-langkah untuk memverifikasi konfigurasi koneksi

Gunakan langkah-langkah berikut untuk memverifikasi konfigurasi koneksi.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Verifikasi konfigurasi koneksi](#).

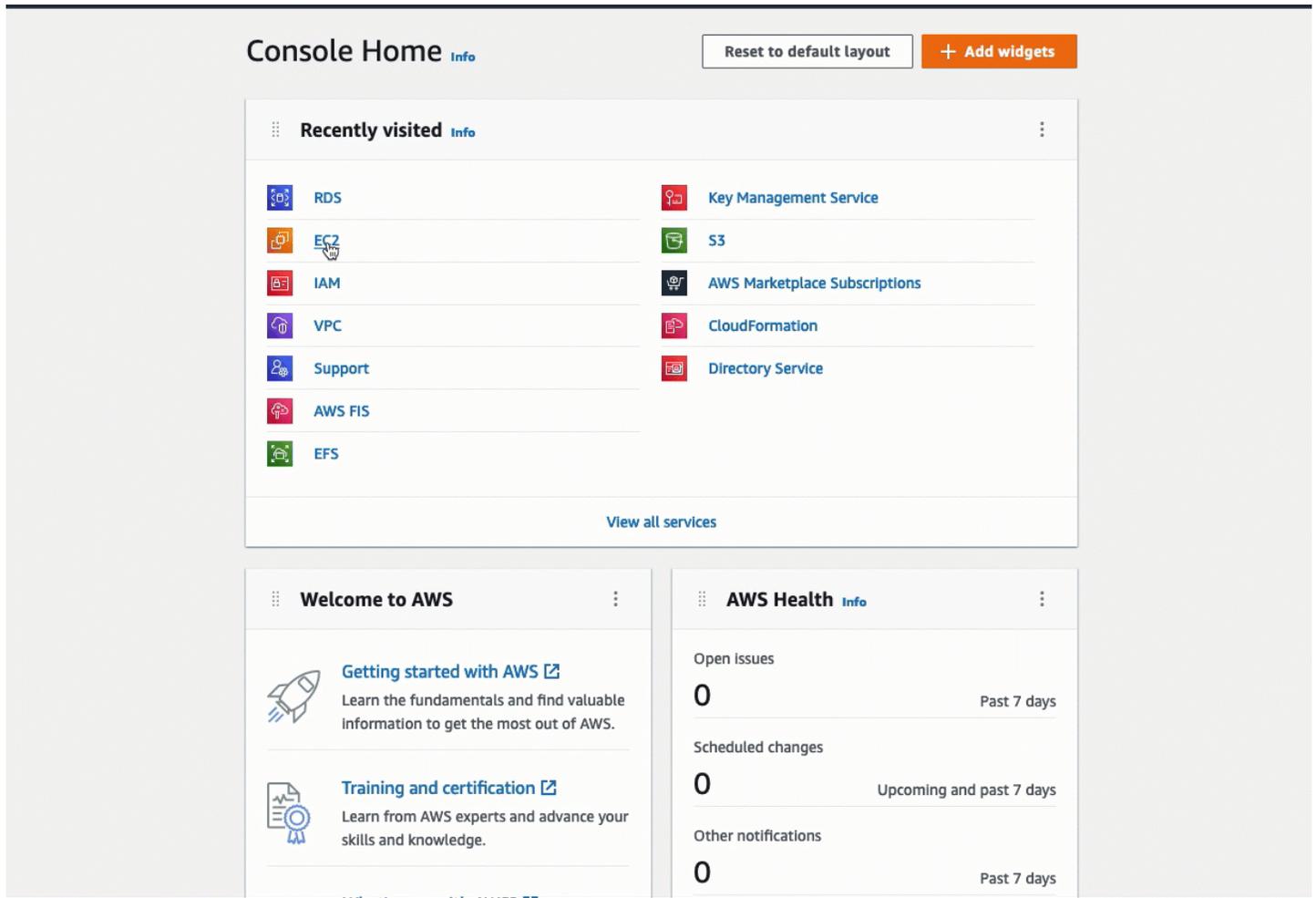
Untuk memverifikasi konfigurasi koneksi menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih instans EC2 yang Anda pilih untuk terhubung ke basis data RDS di tugas sebelumnya, dan pilih tab Keamanan.
4. Pada Detail keamanan, Grup keamanan, verifikasi bahwa grup keamanan yang disebut `ec2-rds-x` ada dalam daftar. `x` adalah angka.
5. Pilih grup keamanan `rds-ec2-x` untuk membukanya.
6. Pilih tab Aturan keluar.
7. 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

8. Buka konsol Amazon RDS di <https://console.aws.amazon.com/rds/>.
9. Di halaman navigasi, pilih Basis Data.
10. Pilih basis data RDS yang Anda buat untuk tutorial ini.
11. Di tab Konektivitas & keamanan, pada Keamanan, Grup keamanan VPC, verifikasi bahwa grup keamanan yang disebut `rds-ec2-x` ditampilkan.
12. Pilih grup keamanan `rds-ec2-x`. Layar Grup Keamanan di konsol EC2 terbuka.
13. Pilih grup keamanan `rds-ec2-x` untuk membukanya.
14. Pilih tab Aturan masuk.
15. 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-langkah sebelumnya.
 - Deskripsi: Aturan untuk mengizinkan koneksi dari instans EC2 dengan **`sg-1234567890example`** terlampir

Dengan memverifikasi bahwa grup keamanan dan aturan grup keamanan ini ada dan bahwa masing-masing ditetapkan ke instans EC2 dan basis data RDS 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.

Opsi 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan meniru fitur koneksi otomatis

Tujuan

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.

Sebelum Anda memulai

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. Jika Anda telah menyelesaikan Opsi 1 dari tutorial ini, Anda sudah memiliki izin 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 untuk menyelesaikan Opsi 3

- [Tugas 1: Luncurkan instans EC2 – opsional](#)
- [Tugas 2: Buat basis data RDS — opsional](#)
- [Tugas 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan membuat grup keamanan dan menyetapkannya ke instans](#)

Tugas 1: Luncurkan instans EC2 – opsional

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 tugas

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 untuk meluncurkan instans EC2

Gunakan langkah-langkah berikut untuk meluncurkan instans EC2 untuk tutorial ini.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Luncurkan instans EC2](#).

Konfigurasi instans EC2

Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **tutorial-instance**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Penetapan otomatis IP publik: Aktif
- 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 AWS Management Console dan buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/](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

Anda dapat menggunakan Amazon EC2 di bawah [tingkat Gratis](#) asalkan AWS akun Anda berusia kurang dari 12 bulan dan Anda memilih jenis `t2.micro` instans (atau `t3.micro` di Wilayah yang tidak `t2.micro` tersedia).

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 bahwa instans Anda menerima alamat IP publik, untuk Tetapkan otomatis IP publik, periksa apakah Aktifkan dipilih. Jika Nonaktifkan dipilih, pilih Edit (di sebelah kanan Pengaturan Jaringan), lalu untuk Tetapkan otomatis IP 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 instans Anda, pada Firewall (grup keamanan), dari daftar tarik turun di samping kotak centang Izinkan 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 [Pemecahan masalah peluncuran instans](#).

Untuk informasi tentang peluncuran instans, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Lihat animasi: Luncurkan instans EC2

The screenshot displays the Amazon EC2 console interface. On the left is a navigation sidebar with categories like EC2 Dashboard, Instances, Images, Elastic Block Store, and Network & Security. The main content area is divided into several sections:

- Resources:** A summary of EC2 resources in the Europe (Stockholm) Region.

Instances (running)	2	Dedicated Hosts	0	Elastic IPs	0
Instances	2	Key pairs	1	Load balancers	0
Placement groups	0	Security groups	10	Snapshots	1
Volumes	3				
- Launch instance:** A section with a prominent orange "Launch instance" button and a "Migrate a server" link. Below it, a note states: "Note: Your instances will launch in the Europe (Stockholm) Region".
- Scheduled events:** A section showing "Europe (Stockholm)" with "No scheduled events".
- Service health:** A section showing the region "Europe (Stockholm)" with a status of "This service is operating normally".
- Zones:** A table listing available zones in the region.

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

Anda sekarang siap untuk [Tugas 2: Buat basis data RDS — opsional](#).

Tugas 2: Buat basis data RDS — opsional

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 tugas

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 untuk membuat basis data RDS

Gunakan langkah-langkah berikut untuk membuat basis data RDS untuk Opsi 3 dari tutorial ini.

Untuk melihat animasi dari langkah-langkah tersebut, lihat [Lihat animasi: Membuat instans DB](#).

Konfigurasi basis data RDS

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.

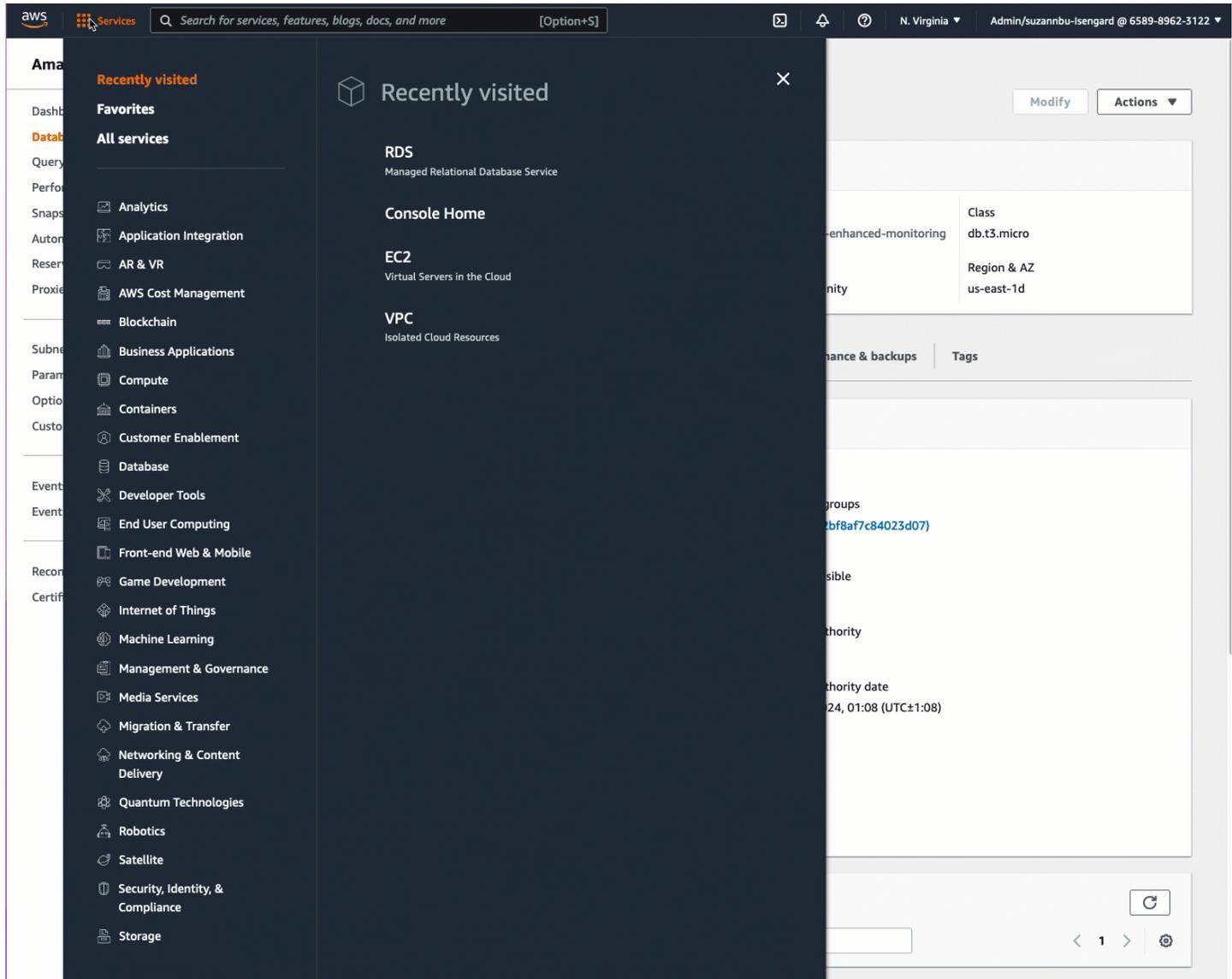
Untuk membuat instans DB MySQL

1. Buka konsol Amazon RDS di <https://console.aws.amazon.com/rds/>.
2. Dari pemilih Region (di kanan atas), pilih Wilayah AWS tempat Anda membuat instans 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



Anda sekarang siap untuk [Tugas 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan membuat grup keamanan dan menyetapkannya ke instans.](#)

Tugas 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan membuat grup keamanan dan menentukannya ke instans

Tujuan tugas

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.

Langkah untuk membuat grup keamanan baru dan menambahkannya ke instans

Gunakan langkah-langkah berikut untuk menghubungkan instans EC2 ke basis data RDS Anda dengan membuat dua grup keamanan baru. Anda 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 Tipe, 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 Tipe, pilih MySQL/Aurora.
 - ii. Untuk Sumber, pilih grup keamanan basis data RDS `rds-ec2-manual-configuration` yang Anda buat di 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 pilih 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 [Bersihkan](#).

Bersihkan

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 dikenakan biaya lebih lanjut.

Topik

- [Akhir instans EC2 Anda](#)
- [Hapus basis data RDS Anda](#)

Akhiri instans EC2 Anda

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.

Hapus basis data RDS Anda

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.

Identifikasi instans EC2 Anda

Anda mungkin perlu menentukan apakah aplikasi Anda berjalan pada instans EC2, terutama jika Anda memiliki lingkungan komputasi campuran. Setiap instance memiliki dokumen identitas instance yang ditandatangani yang dapat Anda verifikasi secara kriptografi. Anda dapat menemukan dokumen-dokumen ini di alamat lokal yang tidak dapat dirutekan berikut. <http://169.254.169.254/latest/dynamic/instance-identity/> Untuk informasi selengkapnya, lihat [Dokumen identitas instans](#).

Periksa sistem UUID

Anda bisa mendapatkan UUID sistem dan melihat oktet awal UUID untuk EC2 (di Linux, ini mungkin huruf kecil). 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 endian kecil, yang tidak termasuk EC2 di awal UUID. Ini mungkin terjadi pada instans EC2 yang menggunakan SMBIOS 2.4 untuk Windows, atau untuk distribusi Linux selain Amazon Linux 2 yang memiliki implementasi SMBIOS sendiri.

Contoh Linux: Dapatkan UUID dari DMI (hanya AMI HVM)

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 pengenalan pembuatan mesin virtual sistem

Pengidentifikasi generasi mesin virtual terdiri dari buffer unik 128-bit yang diartikan sebagai pengidentifikasi integer acak kriptografi. Anda dapat mengambil pengenalan 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 diklona, disalin, atau diimpor ke AWS, seperti dengan [VM Import/Export](#).

Contoh: Ambil pengenalan generasi mesin virtual dari Linux

Anda dapat menggunakan perintah berikut untuk mengambil pengenalan 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.ds1
```

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 pengenalan 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
      })
    }
  }
}
}

```

- (Opsional) Tingkatkan izin terminal Anda untuk langkah-langkah yang tersisa dengan perintah berikut:

```
sudo -s
```

- Gunakan perintah berikut untuk menyimpan ruang alamat yang dikumpulkan sebelumnya:

```
VMGN_ADDR=0xFED01000
```

- 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
```

- Ambil pengidentifikasi pembuatan mesin virtual dari file output dengan perintah berikut:

```
cat vmgenid ; echo
```

Output Anda harus serupa dengan berikut:

```
EC2F335D979132C4165896753E72BD1C
```

Contoh: Ambil pengenalan generasi 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.

Mengelola setelan sistem untuk instans Amazon EC2 Anda

Setelah meluncurkan instans, Anda dapat masuk sebagai administrator untuk membuat perubahan. Bagian ini difokuskan pada pengelolaan pengaturan sistem untuk instans Anda.

Daftar Isi

- [Atur waktu untuk instans Amazon EC2](#)
- [Kontrol status prosesor untuk instans Amazon EC2 Linux Anda](#)
- [Mengoptimalkan opsi CPU](#)
- [AMD SEV-SNP di Amazon EC2](#)
- [Tambahkan komponen sistem Windows menggunakan media instalasi](#)
- [Mengelola pengguna sistem pada instans Linux Anda](#)
- [Tetapkan kata sandi Administrator Windows untuk instans Anda](#)

Atur waktu untuk instans Amazon EC2

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 mengakibatkan perbedaan antara tanggal dalam tanda tangan dan tanggal permintaan, yang menyebabkan AWS penolakan permintaan Anda.

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 yang terhubung dengan satelit dan atom di masing-masing Wilayah AWS untuk memberikan pembacaan waktu yang akurat dan terkini dari standar global Coordinated Universal Time (UTC).

Layanan Amazon Time Sync menggunakan Protokol Waktu Jaringan (NTP), atau menyediakan jam perangkat keras Protokol Waktu Presisi (PTP) lokal [instans yang didukung](#). Jam perangkat keras PTP mendukung baik NTP atau koneksi PTP langsung. Koneksi NTP dan PTP langsung menggunakan sumber waktu yang sangat akurat yang sama, tetapi koneksi PTP langsung lebih akurat daripada koneksi NTP. Koneksi NTP ke Amazon Time Sync Service mendukung leap smearing sementara koneksi PTP ke jam perangkat keras PTP tidak merusak waktu. Untuk informasi selengkapnya, lihat [Detik kabisat](#).

Untuk performa terbaik, sebaiknya gunakan Layanan Sinkronisasi Waktu Amazon lokal pada instans EC2 Anda. Untuk cadangan ke Layanan Sinkronisasi Waktu Amazon lokal pada instans Anda, dan 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 atom dan terhubung satelit kami di masing-masing Wilayah AWS

Topik

- [Atur instans Anda untuk menggunakan Layanan Amazon Time Sync](#)
- [Setel instans Anda atau perangkat apa pun yang terhubung ke internet untuk menggunakan Layanan Amazon Time Sync publik](#)
- [Bandingkan stempel waktu untuk instans Linux Anda](#)

- [Ubah zona waktu instans Anda](#)
- [Detik kabisat](#)
- [Sumber daya terkait](#)

Atur instans Anda untuk menggunakan Layanan Amazon Time Sync

Instans Anda dapat mengakses Layanan Amazon Time Sync lokal sebagai berikut:

- Melalui NTP di titik akhir alamat IP berikut ini:
 - IPv4: 169.254.169.123
 - IPv6: fd00:ec2::123 (Hanya dapat diakses oleh [instans yang dibangun di atas Sistem AWS Nitro.](#))
- (Hanya Linux) Melalui koneksi PTP langsung untuk terhubung ke jam perangkat keras PTP lokal:
 - PHC0

AMI Amazon Linux, AMI Windows, dan sebagian besar AMI mitra mengonfigurasi instans Anda untuk menggunakan titik akhir NTP IPv4 secara default. Ini adalah pengaturan yang disarankan untuk sebagian besar beban kerja pelanggan. Konfigurasi lebih lanjut tidak diperlukan untuk instans yang diluncurkan dari AMI ini kecuali Anda ingin menggunakan titik akhir IPv6 atau terhubung langsung ke jam perangkat keras PTP.

Koneksi NTP dan PTP tidak memerlukan perubahan konfigurasi VPC apa pun, dan instans Anda tidak memerlukan akses ke internet.

Note

Hanya instance Linux yang dapat menggunakan koneksi PTP langsung untuk terhubung ke jam perangkat keras PTP lokal. Instans Windows menggunakan NTP untuk terhubung ke jam perangkat keras PTP lokal.

Topik

- [Hubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync](#)
- [Hubungkan ke titik akhir IPv6 pada Layanan Amazon Time Sync](#)
- [Terhubung ke jam perangkat keras PTP](#)

Hubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync

Bagian ini menjelaskan cara mengonfigurasi instans Anda untuk menggunakan Layanan Amazon Time Sync lokal melalui titik akhir IPv4.

Gunakan instruksi untuk sistem operasi instans Anda.

Linux

AL2023 serta versi terbaru AMI Amazon Linux 2 dan Amazon Linux dikonfigurasi untuk menggunakan titik akhir IPv4 Layanan Amazon Time Sync secara default. Tidak diperlukan konfigurasi lebih lanjut untuk instance yang diluncurkan dari AMI ini dan Anda dapat melewati prosedur berikut.

Jika Anda menggunakan AMI yang tidak memiliki Layanan Amazon Time Sync yang dikonfigurasi secara default, gunakan salah satu prosedur berikut untuk mengonfigurasi Layanan Amazon Time Sync di instans Anda menggunakan klien `chrony`. Tindakan ini membutuhkan penambahan entri server untuk Layanan Amazon Time Sync ke file konfigurasi `chrony`.

Gunakan instruksi untuk sistem operasi instans Anda.

Amazon Linux

Untuk menghubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync di Amazon Linux menggunakan `chrony`

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 (seperti `vim` atau `nano`). Pastikan bahwa file tersebut menyertakan baris berikut:

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

Jika baris tersebut ada, maka Layanan Amazon Time Sync sudah dikonfigurasi untuk menggunakan titik akhir IPv4 dari Layanan Amazon Time Sync dan Anda dapat melanjutkan

ke langkah berikutnya. Jika tidak, tambahkan baris setelah server yang lain atau pernyataan pool yang sudah ada di file, dan simpan perubahan Anda.

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
```

```
210 Number of sources = 7

    .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
    /  .-- Source state '*' = current synced, '+' = combined , '-' = not
combined,
    | /   '?' = unreachable, 'x' = time may be in error, '~' = time too
variable.
    ||                                     .- xxxx [ yyyy ] +/-
zzzz
    ||      Reachability register (octal) -.      |  xxxx = adjusted
offset,
    ||      Log2(Polling interval) --.      |      |  yyyy = measured
offset,
    ||                                     \      |      |  zzzz = estimated
error.
    ||                                     |      |      \
```

MS Name/IP address	Stratum	Poll	Reach	LastRx	Last sample
===== ^* 169.254.169.123 287us	3	6	17	43	-30us[-226us] +/-
^- ec2-12-34-231-12.eu-west> 11ms	2	6	17	43	-388us[-388us] +/-
^- tshirt.heanet.ie 1959us	1	6	17	44	+178us[+25us] +/-
^? tbag.heanet.ie 0ns	0	6	0	-	+0ns[+0ns] +/-
^? bray.walcz.net 0ns	0	6	0	-	+0ns[+0ns] +/-
^? 2a05:d018:c43:e312:ce77:> 0ns	0	6	0	-	+0ns[+0ns] +/-
^? 2a05:d018:dab:2701:b70:b> 0ns	0	6	0	-	+0ns[+0ns] +/-

Dalam output yang ditampilkan, ^* menunjukkan sumber waktu pilihan.

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 Nov 22 13:18:34 2017
System time      : 0.000000626 seconds slow of NTP time
Last offset      : +0.002852759 seconds
RMS offset       : 0.002852759 seconds
Frequency        : 1.187 ppm fast
Residual freq    : +0.020 ppm
Skew             : 24.388 ppm
Root delay       : 0.000504752 seconds
Root dispersion  : 0.001112565 seconds
Update interval  : 64.4 seconds
Leap status      : Normal
```

Ubuntu

Untuk menghubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync di Ubuntu menggunakan chrony

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
```

```
210 Number of sources = 7
```

```
    .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
   /  .- Source state '*' = current synced, '+' = combined , '-' = not
combined,
| /   '?' = unreachable, 'x' = time may be in error, '~' = time too
variable.
```

```

|| .- xxxx [ yyyy ]
+/- zzzz
|| Reachability register (octal) -. | xxxx =
adjusted offset,
|| Log2(Polling interval) --. | | yyyy =
measured offset,
|| \ | | zzzz =
estimated error.
|| | | \
MS Name/IP address Stratum Poll Reach LastRx Last sample

=====
^* 169.254.169.123 3 6 17 12 +15us[ +57us]
+/- 320us
^- tbag.heanet.ie 1 6 17 13 -3488us[-3446us]
+/- 1779us
^- ec2-12-34-231-12.eu-west- 2 6 17 13 +893us[ +935us]
+/- 7710us
^? 2a05:d018:c43:e312:ce77:6 0 6 0 10y +0ns[ +0ns]
+/- 0ns
^? 2a05:d018:d34:9000:d8c6:5 0 6 0 10y +0ns[ +0ns]
+/- 0ns
^? tshirt.heanet.ie 0 6 0 10y +0ns[ +0ns]
+/- 0ns
^? bray.walcz.net 0 6 0 10y +0ns[ +0ns]
+/- 0ns

```

Dalam output yang ditampilkan, baris yang dimulai dengan `^*` menunjukkan sumber waktu pilihan.

5. Verifikasi metrik sinkronisasi waktu yang dilaporkan oleh `chronyc`.

```
ubuntu:~$ chronyc tracking
```

```

Reference ID      : 169.254.169.123 (169.254.169.123)
Stratum          : 4
Ref time (UTC)   : Wed Nov 29 07:41:57 2017
System time      : 0.000000011 seconds slow of NTP time
Last offset      : +0.000041659 seconds
RMS offset       : 0.000041659 seconds
Frequency        : 10.141 ppm slow
Residual freq    : +7.557 ppm
Skew             : 2.329 ppm

```

```
Root delay      : 0.000544 seconds
Root dispersion : 0.000631 seconds
Update interval : 2.0 seconds
Leap status     : Normal
```

SUSE Linux

Dimulai dengan SUSE Linux Enterprise Server 15, `chrony` adalah implementasi default NTP.

Untuk menghubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync di SUSE Linux menggunakan `chrony`

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 kolom 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 melewati prosedur berikut.

Jika Anda menggunakan AMI yang tidak memiliki Layanan Sinkronisasi Waktu Amazon yang dikonfigurasi secara default, pertama-tama verifikasi 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 protokol waktu jaringan (NTP) bawaan untuk AMI Amazon Windows

Amazon Machine Images (AMI) umumnya mematuhi out-of-the-box default kecuali dalam kasus di mana 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 terhadap akurasi. AWS mengonfigurasi interval pembaruan untuk terjadi setiap dua menit sekali.
- Server NTP – Sejak rilis Agustus 2018, AMI menggunakan Layanan Amazon Time Sync secara default. Layanan kali 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 menyediakan waktu ke sistem operasi.
- Interval Poll Khusus - Memeriksa Server NTP yang dikonfigurasi setiap 900 detik (15 menit).

Jalur registri	Nama kunci	Data
HKLM:\System\CurrentControlSet\services\w32time\Config	UpdateInterval	120
HKLM:\System\CurrentControlSet\services\w32time\Parameter	NtpServer	169.254.169.123,0x9
HKLM:\System\CurrentControlSet\services\w32time\Parameter	Tipe	NTP
HKLM:\System\CurrentControlSet\services\w32time\TimeProviders NtpClient	Aktif	1
HKLM:\System\CurrentControlSet\services\w32time\TimeProviders NtpClient	InputProvider	1
HKLM:\System\CurrentControlSet\services\w32time\TimeProviders NtpClient	SpecialPollInterval	900

Hubungkan ke titik akhir IPv6 pada Layanan Amazon Time Sync

Bagian ini menjelaskan perbedaan langkah-langkah yang dijelaskan dalam [Hubungkan ke titik akhir IPv4 pada Layanan Amazon Time Sync](#) 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 [instans yang dibangun di atas Sistem Nitro](#) AWS.

Note

Kami tidak menyarankan untuk 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.

Gunakan instruksi untuk sistem operasi instans Anda.

Linux

Bergantung pada distribusi Linux yang Anda gunakan, ketika Anda mencapai langkah untuk mengedit file `chrony.conf`, 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 pastikan bahwa `chrony` menggunakan titik akhir IPv6 `fd00:ec2::123` untuk menyinkronkan waktu.

```
[ec2-user ~]$ chronyc sources -v
```

Dalam output, jika Anda melihat titik akhir `fd00:ec2::123` IPv6, konfigurasi selesai.

Windows

Saat Anda mencapai langkah untuk mengubah server NTP untuk menggunakan Layanan Sinkronisasi Waktu Amazon, 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
```

Verifikasi bahwa pengaturan baru Anda menggunakan titik akhir `fd00:ec2::123` IPv6 untuk menyinkronkan waktu:

```
w32tm /query /configuration
```

Dalam output, verifikasi yang `NtpServer` menampilkan titik akhir `fd00:ec2::123` IPv6.

Terhubung ke jam perangkat keras PTP

Jam perangkat keras PTP adalah bagian dari [AWS System Nitro](#), sehingga dapat diakses langsung pada [instans EC2 bare metal dan virtual yang didukung](#) tanpa menggunakan sumber daya pelanggan apa pun.

Titik akhir NTP ke jam perangkat keras PTP sama dengan koneksi Layanan Amazon Time Sync biasa melalui IPv4 atau IPv6. Jika perangkat lunak Anda dikonfigurasi ke titik akhir NTP dan berjalan pada instans dengan jam perangkat keras PTP, perangkat lunak itu akan terhubung ke jam perangkat keras PTP secara otomatis melalui NTP.

Persyaratan

Jam perangkat keras PTP tersedia pada instans ketika persyaratan berikut terpenuhi:

- Didukung Wilayah AWS: AS Timur (Virginia N.) dan Asia Pasifik (Tokyo)
- Keluarga instans yang didukung:
 - Tujuan umum: m7a, m7g, m7gd, m7i
 - Komputasi dioptimalkan: C7a, C7gd, C7i
 - Memori dioptimalkan: R7a, R7g, R7gd, R7i
- (Hanya Linux) driver ENA versi 2.10.0 atau yang lebih baru diinstal pada sistem operasi yang didukung. Untuk informasi selengkapnya tentang sistem operasi yang didukung, lihat [prasyarat driver di GitHub](#)

Gunakan instruksi untuk sistem operasi instans Anda.

Linux

Bagian ini menjelaskan cara mengonfigurasi instans Anda untuk menggunakan Layanan Amazon Time Sync lokal melalui jam perangkat keras PTP dengan menggunakan koneksi PTP langsung.

Konfigurasi ini membutuhkan penambahan entri server untuk jam perangkat keras PTP ke file konfigurasi `chrony`.

Jika instans Anda memiliki jam perangkat keras PTP dan Anda mengonfigurasi koneksi NTP (baik ke titik akhir IPv4 maupun IPv6), waktu instans Anda secara otomatis bersumber dari jam perangkat keras PTP. Langkah-langkah di bawah ini mengonfigurasi koneksi PTP langsung, yang akan memberi Anda waktu yang lebih akurat daripada koneksi NTP.

Untuk terhubung ke jam perangkat keras PTP

1. Hubungkan dengan instans Anda dan instal driver kernel Linux untuk Elastic Network Adapter (ENA) versi 2.10.0 atau setelahnya. Untuk petunjuk penginstalan, lihat [Driver kernel Linux untuk keluarga Elastic Network Adapter \(ENA\) GitHub](#).
2. Pastikan bahwa perangkat `/dev/ptp0` muncul di instans Anda.

```
[ec2-user ~]$ ls /dev/ptp0
```

Berikut adalah output yang diharapkan. Jika `/dev/ptp0` tidak ada dalam output, driver ENA tidak akan diinstal dengan benar. Tinjau langkah 1 dalam prosedur ini untuk menginstal driver.

```
/dev/ptp0
```

3. Edit `/etc/chrony.conf` menggunakan editor teks dan tambahkan baris berikut di mana saja di file.

```
refclock PHC /dev/ptp0 poll 0 delay 0.000010 prefer
```

4. Mulai ulang `chrony` menggunakan perintah berikut.

```
[ec2-user ~]$ sudo systemctl restart chronyd
```

5. Verifikasi bahwa `chrony` menggunakan jam perangkat keras PTP untuk menyinkronkan waktu pada instans ini.

```
[ec2-user ~]$ chronyc sources
```

Output yang diharapkan

```
MS Name/IP address          Stratum Poll Reach LastRx Last sample
```

```
=====
#* PHC0                0  0  377  1  +2ns[ +1ns] +/-  5031ns
```

Dalam output yang dihasilkan, * menunjukkan sumber waktu pilihan. PHC0 sesuai dengan jam perangkat keras PTP. Anda mungkin perlu menunggu beberapa detik setelah memulai ulang kroni sebelum tanda bintang muncul.

Windows

Instans Windows hanya mendukung koneksi NTP ke jam perangkat keras PTP lokal.

Titik akhir NTP ke jam perangkat keras PTP sama dengan koneksi Layanan Amazon Time Sync biasa melalui IPv4 atau IPv6. Jika perangkat lunak Anda dikonfigurasi untuk terhubung ke titik akhir NTP dan berjalan pada instans dengan jam perangkat keras PTP, perangkat lunak itu akan secara otomatis terhubung ke jam perangkat keras PTP melalui NTP.

Setel instans Anda atau perangkat apa pun yang terhubung ke internet untuk menggunakan Layanan Amazon Time Sync 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 kinerja 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 Sinkronisasi 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 stempel waktu pada instans ClockBound Amazon EC2 Linux untuk menentukan waktu sebenarnya dari suatu peristiwa. ClockBound mengukur akurasi jam instans EC2 Anda, dan memungkinkan Anda untuk memeriksa apakah stempel waktu yang diberikan ada 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 pustaka open source. Untuk mempelajari selengkapnya ClockBound, termasuk petunjuk penginstalan, 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 kesalahan terikat 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 kesalahan jam perangkat keras PTP terikat

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 kesalahan jam perangkat keras PTP terikat, 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 kesalahan jam yang benar yang terikat pada titik waktu tertentu saat menggunakan koneksi PTP langsung ke jam perangkat keras PTP, Anda harus menambahkan kesalahan jam yang terikat dari chrony atau ClockBound pada saat chrony polling jam perangkat keras PTP. Untuk informasi selengkapnya tentang mengukur dan memantau akurasi jam, lihat [Mengelola akurasi jam instans Amazon EC2 menggunakan Layanan Sinkronisasi Waktu Amazon dan Amazon CloudWatch — Bagian 1](#).

Ubah zona waktu instans Anda

Instans Amazon EC2 disetel 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 pada instans AL2023 atau Amazon Linux 2

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
```

- (Opsional) Konfirmasikan bahwa zona waktu saat ini diperbarui ke zona waktu baru dengan menjalankan perintah `timedatectl` lagi.

```
[ec2-user ~]$ timedatectl
```

Untuk mengubah zona waktu pada instans Amazon Linux

- Identifikasi zona waktu yang akan digunakan pada instans. Direktori `/usr/share/zoneinfo` berisi hierarki file data zona waktu. Jelajahi struktur direktori di lokasi itu untuk menemukan file untuk zona waktu Anda.

```
[ec2-user ~]$ ls /usr/share/zoneinfo
Africa      Chile      GB         Indian     Mideast    posixrules US
America     CST6CDT   GB-Eire    Iran       MST         PRC        UTC
Antarctica  Cuba      GMT        iso3166.tab MST7MDT    PST8PDT    WET
Arctic      EET       GMT0       Israel     Navajo     right      W-SU
...
```

Beberapa entri di lokasi ini berupa direktori (seperti `America`), dan direktori ini berisi file zona waktu untuk kota tertentu. Temukan kota Anda (atau kota di zona waktu Anda) yang akan digunakan untuk instans.

- Perbarui file `/etc/sysconfig/clock` dengan zona waktu baru. Dalam contoh ini, kami menggunakan file data zona waktu untuk Los Angeles, `/usr/share/zoneinfo/America/Los_Angeles`.
 - Buka file `/etc/sysconfig/clock` menggunakan editor teks (seperti `vim` atau `nano`). Anda perlu menggunakan `sudo` dengan perintah editor Anda karena `/etc/sysconfig/clock` dimiliki oleh `root`.

```
[ec2-user ~]$ sudo nano /etc/sysconfig/clock
```

- Temukan entri `ZONE`, dan ubah ke file zona waktu (dengan menghapus bagian `/usr/share/zoneinfo` dari jalur). Misalnya, untuk mengubah ke zona waktu Los Angeles, ubah entri `ZONE` menjadi berikut ini:

```
ZONE="America/Los_Angeles"
```

 Note

Jangan ubah entri `UTC=true` ke nilai lain. Entri ini untuk jam perangkat keras, dan tidak perlu disesuaikan saat Anda menyetel zona waktu yang berbeda pada instans Anda.

- c. Simpan file dan keluar dari editor teks.
3. Buat tautan simbolis di antara `/etc/localtime` dan file zona waktu, sehingga instans menemukan file zona waktu saat merujuk informasi waktu lokal.

```
[ec2-user ~]$ sudo ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
```

4. Reboot sistem untuk mengambil informasi zona waktu baru di semua layanan dan aplikasi.

```
[ec2-user ~]$ sudo reboot
```

5. (Opsional) Konfirmasikan bahwa zona waktu saat ini diperbarui ke zona waktu baru dengan menjalankan perintah `date`. Zona waktu saat ini muncul di output. Dalam contoh berikut, zona waktu saat ini adalah PDT, yang mengacu pada zona waktu Los Angeles.

```
[ec2-user ~]$ date  
Sun Aug 16 05:45:16 PDT 2020
```

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. Tetapkan ke zona waktu lain dengan menggunakan perintah berikut ini:

```
tzutil /s "Pacific Standard Time"
```

Zona waktu baru akan segera berlaku.

Note

Anda dapat menetapkan zona UTC dengan menggunakan perintah berikut ini:

```
tzutil /s "UTC"
```

Untuk mencegah zona waktu Anda berubah setelah Anda mengaturnya untuk Windows Server

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 registri RealTimeIsUniversal telah diatur, lihat langkah 4 dalam prosedur berikut ini. 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. Jika Anda menggunakan AMI Windows Server 2008 (bukan Windows Server 2008 R2) yang dibuat sebelum 22 Februari 2013, kami sarankan memperbarui ke AMI AWS Windows terbaru.

Jika Anda menggunakan AMI yang menjalankan Windows Server 2008 R2 (bukan Windows Server 2008), Anda harus memverifikasi bahwa perbaikan terbaru Microsoft [KB2922223](#) telah diinstal. Jika perbaikan terbaru ini tidak diinstal, kami sarankan memperbarui ke AMI AWS Windows terbaru.

4. (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          0xffffffffc4
DaylightName        REG_SZ              @tzres.dll, -211
DaylightStart       REG_BINARY          0000030002000200000000000000000000
StandardBias        REG_DWORD          0x0
StandardName        REG_SZ              @tzres.dll, -212
StandardStart       REG_BINARY          00000B0001000200000000000000000000
TimeZoneKeyName     REG_SZ              Pacific Standard Time
DynamicDaylightTimeDisabled REG_DWORD          0x0
ActiveTimeBias      REG_DWORD          0x1a4
RealTimeIsUniversal REG_DWORD          0x1
```

Detik kabisat

Detik kabisat, diperkenalkan pada tahun 1972, merupakan penyesuaian satu detik sesekali pada waktu UTC untuk memperhitungkan penyimpangan dalam rotasi bumi, untuk mengakomodasi perbedaan antara Waktu Atom Internasional (KAI) 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. Sumber waktu leap-smear dan detik kabisat adalah 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.

Sumber daya terkait

- AWS Compute Blog: [Sudah Saatnya: Jam Akurat Mikrodetik di Instans Amazon EC2](#)
- (Linux) <https://chrony-project.org/>
- (Windows) [Cara Kerja Layanan Waktu Windows](#) (Microsoft)
- (Windows) [W32tm \(Microsoft\)](#)
- (Windows) [Bagaimana layanan Windows Time memperlakukan detik kabisat](#) (Microsoft)
- (Windows) [Kisah seputar Leap Seconds dan Windows: Kemungkinan bukan Y2K](#) (Microsoft)

Kontrol status prosesor untuk instans Amazon EC2 Linux Anda

Status-C mengontrol tingkat tidur yang dapat dimasuki inti saat idle. Status-C diberi nomor mulai dengan C0 (status paling dangkal di mana inti benar-benar terjaga dan menjalankan instruksi) hingga C6 (keadaan idle terdalam di mana inti dimatikan).

P-state mengontrol performa yang diinginkan (dalam frekuensi CPU) dari sebuah inti. Status-P diberi nomor mulai dari P0 (pengaturan performa tertinggi di mana inti diizinkan untuk menggunakan Intel Turbo Boost Technology untuk meningkatkan frekuensi jika memungkinkan), dan mereka beralih dari P1 (status-P yang meminta frekuensi acuan maksimum) ke P15 (frekuensi serendah mungkin).

Status-C dan Status-P

Tipe instans berikut memberikan kemampuan bagi sistem operasi untuk mengontrol status-C dan status-P prosesor:

- Tujuan umum: `m4.10xlarge` | `m4.16xlarge` | `m5.metal` | `m5d.metal` | `m5n.metal` | `m5zn.metal` | `m6i.metal` | `m6id.metal` | `m7a.metal-48x1` | `m7i.metal-24x1` | `m7i.metal-48x1`
- Komputasi dioptimalkan: `c4.8xlarge` | `c5.metal` | `c5an.metal` | `c5adn.metal` | `c5n.metal` | `c6i.metal` | `c6id.metal` | `c7a.metal-48x1` | `c7i.metal-24x1` | `c7i.metal-48x1`

- Memori dioptimalkan: r4.8xlarge r4.16xlarge r5.metal r5b.metal | r5d.metal | r6i.metal | r7a.metal-48xl | r7i.metal-24xl r7i.metal-48xl | r7iz.metal-16xl | r7iz.metal-32xl | u-6tb1.metal | u-9tb1.metal | u-12tb1.metal | u-18tb1.metal | u-24tb1.metal | x1.16xlarge | x1.32xlarge | x1e.8xlarge | x1e.16xlarge | x1e.32xlarge | z1d.metal
- Penyimpanan yang dioptimalkan: d2.8xlarge | d3.metal | d3en.metal | i3.8xlarge | i3.16xlarge | i3.metal | i3en.metal | h1.8xlarge | h1.16xlarge
- Komputasi yang dipercepat: f1.16xlarge | g3.16xlarge | g4dn.metal | p2.16xlarge | p3.16xlarge

Status-C saja

Tipe instans berikut memberikan kemampuan bagi sistem operasi untuk mengontrol status-C prosesor:

- Tujuan umum: m5.12xlarge m5.24xlarge | m5d.12xlarge | m5d.24xlarge m5n.12xlarge | m5n.24xlarge | m5dn.12xlarge | m5dn.24xlarge | m6a.24xlarge m6a.48xlarge | m6ad.metal | m6i.16xlarge | m6i.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
- Komputasi 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 | 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
- Memori dioptimalkan: r5.12xlarge r5.24xlarge r5d.12xlarge r5d.24xlarge | r5n.12xlarge | r5n.24xlarge | r5dn.12xlarge | r5dn.24xlarge | r6a.24xlarge | r6a.48xlarge | r6i.16xlarge | r6i.32xlarge | r6id.32xlarge | 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 | u-6tb1.56xlarge | u-6tb1.112xlarge | u-9tb1.112xlarge |
u-12tb1.112xlarge | u-18tb1.112xlarge | u-24tb1.112xlarge | u7i-12tb.224xlarge
| u7in-16tb.224xlarge | u7in-24tb.224xlarge | u7in-32tb.224xlarge | z1d.6xlarge
| z1d.12xlarge

- Penyimpanan yang dioptimalkan: d3en.12xlarge | d11.24xlarge | i3en.12xlarge |
i3en.24xlarge | i4i.metal | r5b.12xlarge | r5b.24xlarge | i4i.16xlarge
- Komputasi dipercepat: d11.24xlarge | g5.24xlarge | g5.48xlarge | g6.24xlarge |
g6.48xlarge | inf1.24xlarge | p3dn.24xlarge | p4d.24xlarge | p4de.24xlarge |
vt1.24xlarge

AWS Prosesor Graviton memiliki mode hemat daya bawaan dan beroperasi pada frekuensi tetap. Oleh karena itu, mereka tidak menyediakan kemampuan untuk sistem operasi untuk mengontrol status-C dan status-P.

Anda mungkin harus mengubah pengaturan status-C atau status-P untuk meningkatkan konsistensi performa prosesor, mengurangi latensi, atau menyelaraskan instans Anda dengan beban kerja tertentu. Pengaturan status-C dan status-P default memberikan kinerja maksimum, yang optimal untuk sebagian besar beban kerja. Namun, jika aplikasi Anda akan mendapatkan keuntungan dari pengurangan latensi dengan biaya frekuensi ssatu inti atau dua inti yang lebih tinggi, atau dari performa yang konsisten pada frekuensi yang lebih rendah, dibandingkan dengan frekuensi Turbo Boost yang melonjak, pertimbangkan untuk bereksperimen dengan pengaturan status-C atau status-P 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 lainnya dengan kernel Linux 3.9 atau yang lebih baru. Untuk informasi lebih lanjut tentang distribusi Linux dan kontrol status prosesor lainnya, lihat dokumentasi khusus sistem Anda.

Mengoptimalkan opsi CPU

Banyak instans Amazon EC2 mendukung multithreading, yang memungkinkan banyak 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.

Note

Setiap vCPU adalah utas inti CPU, kecuali untuk instans T2, instans M7a, instans Apple silicon Mac, dan platform ARM 64-bit seperti instans yang didukung oleh prosesor AWS Graviton.

Dalam kebanyakan kasus, ada tipe instans Amazon EC2 yang memiliki kombinasi memori dan jumlah vCPU agar sesuai dengan beban kerja Anda. Namun, Anda dapat menentukan opsi CPU berikut untuk mengoptimalkan instans Anda untuk beban kerja atau kebutuhan bisnis tertentu:

- 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 inti: Anda dapat menonaktifkan multithreading dengan menentukan satu thread per inti CPU. Anda dapat melakukannya untuk beban kerja tertentu, seperti beban kerja komputasi performa tinggi (HPC).

Anda dapat menentukan opsi CPU ini selama peluncuran instan. Tidak ada tambahan atau pengurangan biaya untuk menentukan opsi CPU. Anda dikenai biaya yang sama seperti instans yang diluncurkan dengan opsi CPU default.

Daftar Isi

- [Aturan untuk menentukan opsi CPU](#)
- [Inti CPU dan thread per inti CPU per tipe instans](#)
- [Menentukan opsi CPU untuk instans Anda](#)
- [Melihat opsi CPU untuk instans Anda](#)

Aturan untuk menentukan opsi CPU

Untuk menentukan opsi CPU untuk instans Anda, perhatikan aturan berikut:

- Anda tidak dapat menentukan opsi CPU untuk instans bare metal.

- Opsi CPU hanya dapat ditentukan selama peluncuran instans dan tidak dapat diubah setelah peluncuran.
- Saat meluncurkan sebuah instans, Anda harus menentukan jumlah inti CPU dan utas per inti dalam permintaan. Untuk contoh permintaan, lihat [Menentukan opsi CPU untuk instans Anda](#).
- 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 [Inti CPU dan thread per inti CPU per tipe instans](#).
- Untuk menonaktifkan multithreading, tentukan satu thread per inti.
- Saat Anda [mengubah tipe instans](#) pada instans yang ada, opsi CPU secara otomatis berubah ke opsi CPU default untuk tipe instans baru.
- Opsi CPU yang ditentukan tidak akan berubah setelah Anda menghentikan, memulai, atau me-reboot sebuah instans.

Inti CPU dan thread per inti CPU per tipe instans

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](#)
- [Instans komputasi performa tinggi](#)

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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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, 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, 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, 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, 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, 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, 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, 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, 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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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, 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, 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, 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, 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, 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, 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, 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, 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, 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, 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

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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r5a.12xlarge	48	24	2	6, 12, 18, 24	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r5ad.24xlarge	96	48	2	12, 18, 24, 36, 48	1, 2
r5b.large	2	1	2	1	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r5d.large	2	1	2	1	1, 2
r5d.xlarge	4	2	2	2	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r5dn.xlarge	4	2	2	1, 2	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r5n.2xlarge	8	4	2	2, 4	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r6a.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
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, 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, 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, 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
r6idn.large	2	1	2	1	1, 2
r6idn.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
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
r6in.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
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
r6id.large	2	1	2	1	1, 2
r6id.xlarge	4	2	2	1, 2	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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, 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, 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, 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, 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
u-3tb1.56xlarge	224	112	2	4, 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	8, 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-6tb1.11 2xlarge	448	224	2	8, 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-9tb1.11 2xlarge	448	224	2	8, 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-12tb1.1 12xlarge	448	224	2	8, 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-18tb1.1 12xlarge	448	224	2	8, 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-24tb1.1 12xlarge	448	224	2	8, 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-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
x1.16xlarge	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
x1.32xlarge	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, 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
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
x1e.32xlarge	128	64	2	4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64	1, 2
z1d.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
z1d.xlarge	4	2	2	1, 2	1, 2
z1d.2xlarge	8	4	2	2, 4	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
d3.2xlarge	8	4	2	2, 4	1, 2
d3.4xlarge	16	8	2	2, 4, 6, 8	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
h1.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
i3.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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
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, 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	
f1.2xlarge	8	4	2	1, 2, 3, 4	1, 2
f1.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
f1.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	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, 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
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
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
p2.xlarge	4	2	2	1, 2	1, 2
p2.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
p2.16xlarge	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
p3.2xlarge	8	4	2	1, 2, 3, 4	1, 2
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
p5.48xlarge	192	96	2	12, 24, 36, 48, 60, 72, 84, 96	1, 2
trn1.2xlarge	8	4	2	2, 4	1, 2
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

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
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
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

Instans komputasi performa tinggi

Jenis 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,	1

Jenis instans	vCPU default	Inti CPU default	Thread default per inti	Inti CPU yang valid	Thread valid per inti
				48, 50, 52, 54, 56, 58, 60, 62, 64	

Menentukan opsi CPU untuk instans Anda

Anda dapat menentukan opsi CPU ini selama peluncuran instan.

[Contoh berikut menjelaskan cara menentukan opsi CPU saat menggunakan wizard instans peluncuran di konsol EC2 dan perintah run-instance, dan halaman template create launch di konsol EC2 dan AWS CLI perintah create-launch-template.](#) AWS CLI Untuk Armada EC2 atau Armada Spot, Anda harus menentukan opsi CPU dalam templat peluncuran.

Contoh berikut adalah untuk tipe instans `r5.4xlarge`, yang memiliki [nilai default](#) berikut:

- Inti CPU default: 8
- Thread default per inti: 2
- vCPUs default: 16 (8 * 2)
- Jumlah core CPU yang valid: 2, 4, 6, 8
- Jumlah thread per inti yang valid: 1, 2

Menonaktifkan multithreading

Untuk menonaktifkan multithreading, tentukan 1 thread per inti.

New console

Untuk menonaktifkan multithreading selama peluncuran instans

1. Ikuti prosedur [Meluncurkan instans dengan cepat](#) dan konfigurasi instans Anda sesuai kebutuhan.
2. Perluas Detail lanjutan, dan pilih kotak centang Tentukan 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 multithreading, pada Thread per inti, pilih 1.
5. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Old console

Untuk menonaktifkan multithreading selama peluncuran instans

1. Ikuti prosedur [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).
2. Di halaman Konfigurasi Detail Instans, pada Opsi CPU, pilih Tentukan 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 multithreading, pada Thread per inti, pilih 1.
5. Lanjutkan seperti yang diminta oleh wizard. Setelah Anda selesai meninjau opsi di halaman Peluncuran Instans Peninjauan, pilih Luncurkan. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).

AWS CLI

Untuk menonaktifkan multithreading selama peluncuran instans

Gunakan perintah [run-instances](#) AWS CLI dan tentukan nilai 1 untuk `ThreadsPerCore` untuk parameter `--cpu-options`. Untuk `CoreCount`, tentukan jumlah inti CPU. Dalam contoh ini, untuk menentukan jumlah inti CPU default untuk instans `r5.4xlarge`, tentukan nilai 8.

```
aws ec2 run-instances \  
  --image-id ami-1a2b3c4d \  
  --instance-type r5.4xlarge \  
  --cpu-options "CoreCount=8,ThreadsPerCore=1" \  
  --key-name MyKeyPair
```

Tentukan jumlah kustom vCPU saat peluncuran

Anda dapat menyesuaikan jumlah inti CPU dan utas per inti untuk instans tersebut.

Contoh berikut meluncurkan `r5.4xlarge` instance dengan 4 vCPU.

New console

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

1. Ikuti prosedur [Meluncurkan instans dengan cepat](#) dan konfigurasi instans Anda sesuai kebutuhan.
2. Perluas Detail lanjutan, dan pilih kotak centang Tentukan opsi CPU.
3. Untuk mendapatkan 4 vCPU, tentukan 2 core CPU dan 2 thread per core, 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Old console

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

1. Ikuti prosedur [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).
2. Di halaman Konfigurasi Detail Instans, pada Opsi CPU, pilih Tentukan opsi CPU.
3. Untuk mendapatkan 4 vCPU, tentukan 2 core CPU dan 2 thread per core, sebagai berikut:
 - Untuk jumlah inti, pilih 2.
 - Untuk Thread per inti, pilih 2.
4. Lanjutkan seperti yang diminta oleh wizard. Setelah Anda selesai meninjau opsi di halaman Peluncuran Instans Peninjauan, pilih Luncurkan. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama](#).

AWS CLI

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

Gunakan AWS CLI perintah [run-instance](#) dan tentukan jumlah core CPU dan jumlah thread dalam parameter. `--cpu-options` Anda dapat menentukan 2 core CPU dan 2 thread per core untuk mendapatkan 4 vCPU.

```
aws ec2 run-instances \  
  --image-id ami-1a2b3c4d \  
  --instance-type r5.4xlarge \  
  --cpu-options "CoreCount=2,ThreadsPerCore=2" \  
  --key-name MyKeyPair
```

Atau, tentukan 4 core CPU dan 1 thread per core (nonaktifkan multithreading) untuk mendapatkan 4 vCPU:

```
aws ec2 run-instances \  
  --image-id ami-1a2b3c4d \  
  --instance-type r5.4xlarge \  
  --cpu-options "CoreCount=4,ThreadsPerCore=1" \  
  --key-name MyKeyPair
```

Tentukan jumlah kustom vCPU dalam templat peluncuran

Anda dapat menyesuaikan jumlah inti CPU dan thread per inti untuk instans dalam templat peluncuran.

Contoh berikut membuat template peluncuran yang menentukan konfigurasi untuk sebuah *r5.4xlarge* instance dengan 4 vCPU.

Console

Untuk menentukan jumlah kustom vCPU dalam templat peluncuran

1. Ikuti prosedur [Buat template peluncuran dari parameter](#) dan konfigurasi templat peluncuran Anda sesuai kebutuhan.
2. Perluas Detail lanjutan, dan pilih kotak centang Tentukan opsi CPU.
3. Untuk mendapatkan 4 vCPU, tentukan 2 core CPU dan 2 thread per core, 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 [Meluncurkan sebuah instans dari templat peluncuran](#).

AWS CLI

Untuk menentukan jumlah kustom vCPU dalam templat peluncuran

Gunakan AWS CLI perintah [create-launch-template](#) dan tentukan jumlah core CPU dan jumlah thread dalam parameter. `CpuOptions` Anda dapat menentukan 2 core CPU dan 2 thread per core 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-7b16de0c"  
  }],  
  "ImageId": "ami-8c1be5f6",  
  "InstanceType": "r5.4xlarge",  
  "TagSpecifications": [{  
    "ResourceType": "instance",  
    "Tags": [{  
      "Key": "Name",  
      "Value": "webserver"  
    }]  
  }],  
  "CpuOptions": {  
    "CoreCount": 2,  
    "ThreadsPerCore": 2  
  }  
}
```

Atau, tentukan 4 core CPU dan 1 thread per core (nonaktifkan multithreading) untuk mendapatkan 4 vCPU:

```
{
```

```
"NetworkInterfaces": [{
  "AssociatePublicIpAddress": true,
  "DeviceIndex": 0,
  "Ipv6AddressCount": 1,
  "SubnetId": "subnet-7b16de0c"
}],
"ImageId": "ami-8c1be5f6",
"InstanceType": "r5.4xlarge",
"TagSpecifications": [{
  "ResourceType": "instance",
  "Tags": [{
    "Key": "Name",
    "Value": "webserver"
  }]
}],
"CpuOptions": {
  "CoreCount": 4,
  "ThreadsPerCore": 1
}
}
```

Melihat opsi CPU untuk instans Anda

Anda dapat melihat opsi CPU untuk instans yang ada di konsol Amazon EC2 atau dengan menjelaskan instans tersebut menggunakan AWS CLI.

Console

Untuk melihat opsi CPU untuk sebuah instans menggunakan konsol

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 instans (AWS CLI)

Gunakan perintah [describe-instances](#).

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

```
...
  "Instances": [
    {
      "Monitoring": {
        "State": "disabled"
      },
      "PublicDnsName": "ec2-198-51-100-5.eu-central-1.compute.amazonaws.com",
      "State": {
        "Code": 16,
        "Name": "running"
      },
      "EbsOptimized": false,
      "LaunchTime": "2018-05-08T13:40:33.000Z",
      "PublicIpAddress": "198.51.100.5",
      "PrivateIpAddress": "172.31.2.206",
      "ProductCodes": [],
      "VpcId": "vpc-1a2b3c4d",
      "CpuOptions": {
        "CoreCount": 34,
        "ThreadsPerCore": 1
      },
      "StateTransitionReason": "",
      ...
    }
  ]
  ...
```

Dalam output yang dikembalikan, bidang `CoreCount` menunjukkan jumlah inti untuk instans tersebut. Bidang `ThreadsPerCore` menunjukkan jumlah thread per inti.

Atau, untuk melihat informasi CPU, Anda dapat terhubung ke instans Anda dan menggunakan salah satu alat sistem berikut:

- Windows Task Manager pada instance Windows Anda
- `lscpu` Perintah pada instance Linux Anda

Anda dapat menggunakan AWS Config untuk merekam, menilai, mengaudit, dan mengevaluasi perubahan konfigurasi untuk instance, termasuk instance yang dihentikan. Untuk informasi selengkapnya, lihat [Memulai AWS Config](#) di Panduan Pengguna AWS Config .

AMD SEV-SNP di Amazon EC2

AMD Secure Encrypted Virtualization-Secure Nested Paging (AMD SEV-SNP) adalah fitur CPU yang menyediakan properti berikut:

- Pengesahan — AMD SEV-SNP memungkinkan Anda untuk mengambil laporan pengesahan yang ditandatangani yang berisi ukuran kriptografi yang dapat digunakan untuk memvalidasi status dan identitas instans, dan bahwa instans tersebut berjalan pada perangkat keras AMD asli. Untuk informasi selengkapnya, lihat [Pengesahan dengan AMD SEV-SNP](#).
- Enkripsi memori — Dimulai dengan prosesor AMD EPYC (Milan), AWS Graviton2, dan Intel Xeon Scalable (Ice Lake), memori instance selalu dienkripsi. Instans yang diaktifkan untuk AMD SEV-SNP menggunakan kunci khusus instans untuk enkripsi memori mereka.

Konsep dan terminologi

Sebelum Anda mulai menggunakan AMD SEV-SNP, pastikan Anda terbiasa dengan konsep dan terminologi berikut.

Laporan pengesahan AMD SEV-SNP

Laporan pengesahan AMD SEV-SNP adalah dokumen yang dapat diminta oleh instans dari CPU. Laporan pengesahan AMD SEV-SNP dapat digunakan untuk memvalidasi status dan identitas suatu instans, 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 pengesahan AMD SEV-SNP ditandatangani dengan tanda tangan VLEK yang kembali ke root kepercayaan AMD.

VLEK

Versioned Loaded Endorsement Key (VLEK) adalah kunci penandatanganan berseri yang disertifikasi oleh AMD dan digunakan oleh CPU AMD untuk menandatangani laporan pengesahan AMD SEV-SNP. 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

Untuk menggunakan AMD SEV-SNP, Anda harus melakukan hal 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 instance Anda di didukung Wilayah AWS. Saat ini, hanya AS Timur (Ohio) dan Eropa (Irlandia) yang didukung.
- Gunakan AMI dengan mode boot `uefi` atau `uefi-preferred` dan sistem operasi yang mendukung AMD SEV-SNP. Untuk informasi lebih lanjut tentang dukungan AMD SEV-SNP pada sistem operasi Anda, lihat dokumentasi sistem operasi masing-masing. Untuk AWS, AMD SEV-SNP didukung pada AL2023, RHEL 9.3, SLES 15 SP4, dan Ubuntu 23.04 dan yang lebih baru.

Pertimbangan

Anda hanya dapat mengaktifkan AMD SEV-SNP saat meluncurkan instans. Ketika AMD SEV-SNP dihidupkan untuk peluncuran instans Anda, aturan berikut berlaku.

- AMD SEV-SNP tidak dapat dimatikan. Itu tetap menyala 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.
- Host Khusus tidak didukung.
- Jika host yang mendasari 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.

Penetapan harga

Saat meluncurkan instans Amazon EC2 dengan AMD SEV-SNP diaktifkan, Anda akan dikenai biaya penggunaan tambahan per jam yang setara dengan 10 persen [tarif per jam Sesuai Permintaan](#) dari tipe instans yang dipilih.

Biaya penggunaan AMD SEV-SNP 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, Anda akan dikenakan biaya penggunaan tambahan per jam yang setara dengan 10 persen [tarif per jam Sesuai Permintaan](#) dari tipe instans yang dipilih. Jika strategi alokasi menggunakan harga sebagai masukan, Armada Spot tidak termasuk biaya tambahan ini; hanya harga Spot yang digunakan.

Bekerja dengan AMD SEV-SNP di Amazon EC2

Selesaikan tugas-tugas berikut untuk bekerja dengan AMD SEV-SNP di Amazon EC2.

Tugas

- [Menemukan tipe instans yang didukung](#)
- [Aktifkan AMD SEV-SNP saat peluncuran](#)
- [Periksa status AMD SEV-SNP](#)

Menemukan tipe instans yang didukung

Anda dapat menggunakan AWS CLI untuk menemukan jenis instance yang mendukung AMD SEV-SNP.

Untuk menemukan jenis instance yang mendukung AMD SEV-SNP menggunakan AWS CLI, gunakan perintah berikut. [describe-instance-types](#)

```
$ C:\> aws ec2 describe-instance-types \
--filters Name=processor-info.supported-features,Values=amd-sev-snp \
--query 'InstanceTypes[*].InstanceType'
```

Contoh keluaran

```
[
  "r6a.2xlarge",
```

```
"m6a.large",  
"m6a.2xlarge",  
"r6a.xlarge",  
"c6a.16xlarge",  
"c6a.8xlarge",  
"m6a.4xlarge",  
"c6a.12xlarge",  
"r6a.4xlarge",  
"c6a.xlarge",  
"c6a.4xlarge",  
"c6a.2xlarge",  
"m6a.xlarge",  
"c6a.large",  
"r6a.large",  
"m6a.8xlarge"  
]
```

Aktifkan AMD SEV-SNP saat peluncuran

Anda dapat menggunakan AWS CLI untuk meluncurkan instance dengan AMD SEV-SNP dihidupkan.

Untuk meluncurkan instance dengan AMD SEV-SNP dihidupkan menggunakan AWS CLI, gunakan [run-instances](#) perintah dan sertakan opsi. `--cpu-options AmdSevSnp=enabled` Untuk `--image-id`, tentukan AMI dengan mode boot `uefi` atau `uefi-preferred` dan sistem operasi yang mendukung AMD SEV-SNP. Untuk `--instance-type`, tentukan tipe instans yang didukung.

```
$ C:\> aws ec2 run-instances \  
--image-id supported_ami_id \  
--instance-type supported_instance_type \  
--key-name key_pair_name \  
--subnet-id subnet_id \  
--cpu-options AmdSevSnp=enabled
```

Periksa status AMD SEV-SNP

Anda dapat menggunakan salah satu metode berikut untuk memeriksa status AMD SEV-SNP.

AWS CLI

Untuk memeriksa apakah AMD SEV-SNP dihidupkan untuk sebuah instance menggunakan AWS CLI, gunakan perintah. [describe-instances](#) Untuk `--instance-ids`, tentukan ID instans yang akan diperiksa.

```
$ C:\> aws ec2 describe-instances --instance-ids instance_id
```

Dalam output perintah, nilai untuk `AmdSevSnp` in `CpuOptions` menunjukkan apakah AMD SEV-SNP dihidupkan atau dimatikan.

AWS CloudTrail

Dalam AWS CloudTrail hal permintaan peluncuran instance, nilai `"cpuOptions"`: `{"AmdSevSnp": enabled}` menunjukkan bahwa AMD SEV-SNP diaktifkan untuk instance tersebut.

Pengesahan dengan AMD SEV-SNP

Pengesahan adalah proses yang memungkinkan instans Anda membuktikan status dan identitasnya. Saat Anda mengaktifkan AMD SEV-SNP untuk instans Anda, Anda dapat meminta laporan pengesahan AMD SEV-SNP dari prosesor dasarnya. Laporan pengesahan AMD SEV-SNP berisi hash kriptografi, yang disebut pengukuran peluncuran, dari konten memori tamu awal dan status vCPU awal. Laporan pengesahan ditandatangani dengan tanda tangan VLEK yang berantai kembali ke root 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.

Untuk melakukan pengesahan dengan AMD SEV-SNP, selesaikan langkah-langkah berikut.

Langkah 1: Dapatkan laporan pengesahan

Pada langkah ini, Anda menginstal dan membangun `snpquest` utilitas, dan kemudian menggunakannya untuk meminta laporan pengesahan AMD SEV-SNP dan sertifikat.

1. Jalankan perintah berikut untuk membangun `snpquest` utilitas dari file [snpquest repository](#).

```
$ C:\> git clone https://github.com/virtee/snpquest.git
$ C:\> cd snpquest
$ C:\> cargo build -r
$ C:\> cd target/release
```

2. Hasilkan permintaan untuk laporan pengesahan. Utilitas meminta laporan pengesahan dari host, dan menuliskannya ke file biner dengan data permintaan yang disediakan.

Contoh berikut membuat string permintaan acak, dan menggunakannya sebagai file permintaan (`request-file.txt`). Ketika perintah mengembalikan laporan pengesahan itu disimpan di jalur file yang Anda tentukan (`report.bin`). Dalam hal ini, utilitas menyimpan laporan di direktori saat ini.

```
$ C:\> ./snpguest report report.bin request-file.txt --random
```

3. Minta sertifikat dari memori host, dan simpan sebagai file PEM. Contoh berikut menyimpan file dalam direktori yang sama dengan `snpguest` utilitas. Jika sertifikat sudah ada di direktori yang ditentukan, sertifikat tersebut akan ditimpa.

```
$ C:\> ./snpguest certificates PEM ./
```

Langkah 2: Validasi tanda tangan laporan pengesahan

Laporan pengesahan ditandatangani dengan sertifikat, yang disebut Versioned Loaded Endorsement Key (VLEK), yang dikeluarkan oleh AMD untuk AWS. Pada langkah ini, Anda dapat memvalidasi bahwa sertifikat VLEK dikeluarkan oleh AMD, dan bahwa laporan pengesahan ditandatangani oleh sertifikat VLEK itu.

1. Unduh root sertifikat kepercayaan VLEK dari situs web resmi AMD ke direktori saat ini.

```
$ C:\> 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.

```
$ C:\> sudo openssl verify --CAfile ./cert_chain.pem vlek.pem
```

Keluaran yang diharapkan

```
certs/vcek.pem: OK
```

3. Gunakan pemanfaatan `snpguest` untuk memvalidasi bahwa laporan pengesahan ditandatangani oleh sertifikat VLEK.

```
$ C:\> ./snpguest verify attestation ./ report.bin
```

Keluaran yang diharapkan

```
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!
```

Tambahkan komponen sistem Windows menggunakan media instalasi

Sistem operasi Windows Server mencakup banyak komponen opsional. Menyertakan semua komponen opsional di setiap AMI Server Windows Amazon EC2 tidaklah praktis. Sebagai gantinya, kami memberi Anda snapshot EBS media instalasi yang memiliki file yang diperlukan untuk mengonfigurasi atau menginstal komponen pada instans Windows Anda.

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 AWS Management Console atau alat baris perintah untuk mendapatkan ID instance dan Availability Zone dari instance Anda. Anda harus membuat volume EBS di Zona Ketersediaan yang sama dengan instans.

Tambahkan komponen Windows menggunakan konsol

Gunakan prosedur berikut untuk menggunakan AWS Management Console untuk menambahkan komponen Windows ke instans Anda.

Untuk menambahkan komponen Windows ke instans Anda menggunakan konsol

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 di instans Amazon EC2](#).
14. Pilih Lampirkan volume.
15. Hubungkan ke instans Anda dan sediakan volume. Untuk informasi selengkapnya, lihat [Menyediakan volume Amazon EBS untuk digunakan](#) dalam 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.

Tambahkan komponen Windows menggunakan Alat untuk Windows PowerShell

Gunakan prosedur berikut untuk menggunakan Alat untuk Windows PowerShell untuk menambahkan komponen Windows ke instans Anda.

Tambahkan komponen Windows ke instans Anda menggunakan Alat untuk Windows PowerShell

1. Gunakan [Get-EC2Snapshot](#) cmdlet dengan `description` filter Owner dan untuk mendapatkan daftar snapshot media instalasi yang tersedia.

```
PS C:\> Get-EC2Snapshot -Owner amazon -Filter @{ Name="description";
  Values="Windows*" }
```

2. Dalam keluaran, catat ID snapshot yang cocok dengan arsitektur sistem dan preferensi bahasa Anda. Sebagai contoh:

```
...
DataEncryptionKeyId :
Description          : Windows 2019 English Installation Media
Encrypted            : False
KmsKeyId             :
OwnerAlias           : amazon
OwnerId              : 123456789012
Progress             : 100%
SnapshotId           : snap-22da283e
StartTime            : 10/25/2019 8:00:47 PM
State                : completed
StateMessage         :
Tags                 : {}
VolumeId             : vol-be5eafcb
VolumeSize           : 6
...
```

3. Gunakan [New-EC2Volume](#) cmdlet untuk membuat volume dari snapshot. Tentukan Zona Ketersediaan yang sama dengan instans Anda.

```
PS C:\> New-EC2Volume -AvailabilityZone us-east-1a -VolumeType gp2 -
  SnapshotId snap-22da283e
```

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-22da283e
State         : creating
Tags          : {}
VolumeId      : vol-06aa9e1fbf8b82ed1
VolumeType    : gp2
```

- Gunakan [Add-EC2Volume](#)cmdlet untuk melampirkan volume ke instance Anda.

```
PS C:\> Add-EC2Volume -InstanceId i-087711ddaf98f9489 -
VolumeId vol-06aa9e1fbf8b82ed1 -Device xvdh
```

- Hubungkan ke instans Anda dan sediakan volume. Untuk informasi selengkapnya, lihat [Menyediakan volume Amazon EBS untuk digunakan](#) dalam Panduan Pengguna Amazon EBS.

Important

Jangan menginisialisasi volume.

- Buka Panel Kontrol, Program dan Fitur. Pilih Aktifkan atau nonaktifkan fitur Windows. Jika Anda diminta untuk media instalasi, tentukan volume EBS dengan media instalasi.
- (Opsional) Ketika Anda selesai dengan media instalasi, gunakan [Dismount-EC2Volume](#)cmdlet untuk melepaskan volume dari instans Anda. Setelah Anda melepaskan volume, Anda dapat menggunakan [Remove-EC2Volume](#)cmdlet untuk menghapus volume.

Tambahkan komponen Windows menggunakan AWS CLI

Gunakan prosedur berikut untuk menggunakan AWS CLI untuk menambahkan komponen Windows ke instans Anda.

Untuk menambahkan komponen Windows ke instans Anda menggunakan AWS CLI

- 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*
```

- 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"
    },
    ...
  ]
}
```

- Gunakan perintah [create-volume](#) untuk membuat volume dari snapshot. Tentukan Zona Ketersediaan yang sama dengan instans Anda.

```
aws ec2 create-volume --snapshot-id snap-22da283e --volume-type gp2 --availability-zone us-east-1a
```

- Pada keluaran, catat ID volume.

```
{
  "AvailabilityZone": "us-east-1a",
  "Encrypted": false,
  "VolumeType": "gp2",
  "VolumeId": "vol-0c98b37f30bcbc290",
  "State": "creating",
  "Iops": 100,
  "SnapshotId": "snap-22da283e",
  "CreateTime": "2017-04-18T10:33:10.940Z",
  "Size": 6
}
```

- 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 [Menyediakan volume Amazon EBS untuk digunakan](#) dalam 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.

Mengelola pengguna sistem pada instans 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 instans.

 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](#). 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](#)
- [Buat pengguna](#)
- [Menghapus pengguna](#)

Nama pengguna default

Nama pengguna default untuk instans EC2 Anda ditentukan oleh AMI yang ditentukan saat Anda meluncurkan instans.

Nama pengguna default adalah:

- Untuk AL2023, Amazon Linux 2, atau AMI Amazon Linux, nama penggunanya adalah `ec2-user`.
- Untuk AMI CentOS, nama penggunanya adalah `centos` atau `ec2-user`.
- Untuk AMI Debian, nama penggunanya adalah `admin`.
- Untuk AMI Fedora, nama penggunanya adalah `fedora` atau `ec2-user`.
- Untuk AMI RHEL, nama penggunanya adalah `ec2-user` atau `root`.
- Untuk AMI SUSE, nama penggunanya adalah `ec2-user` atau `root`.
- Untuk AMI Ubuntu, nama penggunanya adalah `ubuntu`.
- Untuk AMI Oracle, 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

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 informasi selengkapnya, lihat [Tip untuk Mengamankan Instans EC2 Anda](#).

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 dengan EC2 Instance Connect](#).

Buat pengguna

Pertama buat pengguna, lalu tambahkan kunci publik SSH yang memungkinkan pengguna untuk terhubung dan masuk ke instans.

Untuk membuat pengguna

1. [Buat pasangan kunci baru](#). Anda harus menyediakan file `.pem` untuk pengguna yang Anda buat penggunaannya. Mereka harus menggunakan file ini untuk terhubung ke instans.
2. Ambil kunci publik dari pasangan kunci yang Anda buat di langkah sebelumnya.

```
$ C:\> 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
AAAAB3NzaC1yc2EAAAADAQABAAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706Vhz2ItxCih
+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/
d6RJhJ0I0iBXRlsLnBItnctkiJ7FbtXJMXLvvwJryDUilBMTjYtwB+QhYXUM0zce5Pjz5/
i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco
+CY/5WtUBkrHmFJr6HcXkvJdWpkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi
+z7wB3RbBQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

3. [Hubungkan dengan instans](#).

- 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*.

- Amazon Linux dan Amazon Linux 2

Dengan Amazon Linux dan Amazon Linux 2, pengguna dibuat dengan autentikasi kata sandi yang 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
```

- 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.

- 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.
 - 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](#).

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
```

Tetapkan kata sandi Administrator Windows untuk instans Anda

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 sebuah instans, Anda akan diminta untuk menentukan akun Administrator dan kata sandi default.

Dengan AWS Windows AMI untuk Windows Server 2012 R2 dan sebelumnya, [Konfigurasi instance Windows menggunakan layanan EC2config \(legacy\)](#) menghasilkan kata sandi default.

Dengan AWS Windows AMI untuk Windows Server 2016 dan 2019, [Konfigurasi instans Windows menggunakan EC2Launch](#) menghasilkan kata sandi default. Dengan AMI AWS Windows untuk Windows Server 2022 dan yang lebih baru [Konfigurasi instans Windows menggunakan EC2Launch v2](#), buat kata sandi default.

Note

Dengan Windows Server 2016 dan lebih baru, `Password never expires` dinonaktifkan untuk administrator lokal. Dengan Windows Server 2012 R2 dan sebelumnya, `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 [Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa](#).

Mengelola driver perangkat untuk instans Amazon EC2

Beberapa driver tidak diinstal sebelumnya pada AMI EC2 yang Anda luncurkan. Orang lain mungkin memerlukan pembaruan untuk memanfaatkan fungsionalitas yang diperluas. Topik berikut mencakup penginstalan, pembaruan, dan konfigurasi untuk beberapa driver perangkat yang dilampirkan ke instans EC2 Anda.

Daftar Isi

- [Instal driver NVIDIA di instans Amazon EC2 Anda](#)
- [Instal driver AMD di instans Amazon EC2 Anda](#)
- [Driver paravirtual untuk instans Windows](#)

- [AWS Driver NVMe untuk instance Windows](#)

Instal driver NVIDIA di instans Amazon EC2 Anda

Instans dengan GPU NVIDIA terpasang, seperti instans P3 atau G4dn, harus menginstal driver NVIDIA yang sesuai. Bergantung pada tipe instans, Anda dapat mengunduh driver NVIDIA publik, mengunduh driver dari Amazon S3 yang hanya tersedia untuk pelanggan AWS, atau menggunakan AMI dengan driver yang telah diinstal sebelumnya.

Untuk menginstal driver AMD pada instance dengan GPU AMD yang terpasang, seperti instance G4ad, lihat [Menginstal driver AMD](#). Untuk menginstal driver NVIDIA, lihat [Menginstal driver NVIDIA](#).

Daftar Isi

- [Tipe driver NVIDIA](#)
- [Driver yang tersedia berdasarkan tipe instans](#)
- [Opsi instalasi](#)
 - [Opsi 1: AMI dengan driver NVIDIA terinstal](#)
 - [Opsi 2: Driver NVIDIA publik](#)
 - [Opsi 3: driver GRID \(instans G6, Gr6, G5, G4dn, dan G3\)](#)
 - [Opsi 4: Driver game NVIDIA \(instans G5 dan G4dn\)](#)
- [Menginstal CUDA versi tambahan](#)

Tipe driver NVIDIA

Berikut ini adalah tipe utama driver NVIDIA yang dapat digunakan dengan instans berbasis GPU.

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 driver Tesla, GRID, dan game

- OpenCL, OpenGL, dan Vulkan
- NVIDIA CUDA dan pustaka terkait (misalnya, cuDNN, TensorRT, nvJPEG, and cuBLAS)
- NVENC untuk encode video dan NVDEC untuk decode video
- API khusus Windows: DirectX, Direct2D, Akselerasi Video DirectX, DirectX Raytracing

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

Jenis instans	Driver Tesla	Driver GRID	Driver game
G6	Ya	Ya	Tidak
Gr6	Ya	Ya	Tidak
P2	Ya	Tidak	Tidak
P3	Ya	Tidak	Tidak
P4d	Ya	Tidak	Tidak
P4de	Ya	Tidak	Tidak

¹ Driver Tesla ini juga mendukung aplikasi grafis yang dioptimalkan khusus untuk platform ARM64

² Menggunakan AMI Marketplace saja

Opsi instalasi

Gunakan salah satu opsi berikut untuk mendapatkan driver NVIDIA yang diperlukan untuk instans GPU Anda.

Opsi

- [Opsi 1: AMI dengan driver NVIDIA terinstal](#)
- [Opsi 2: Driver NVIDIA publik](#)
- [Opsi 3: driver GRID \(instans G6, Gr6, G5, G4dn, dan G3\)](#)
- [Opsi 4: Driver game NVIDIA \(instans G5 dan G4dn\)](#)

Opsi 1: AMI dengan driver NVIDIA terinstal

AWS dan NVIDIA menawarkan Gambar Mesin Amazon (AMI) yang berbeda yang disertakan dengan driver NVIDIA yang diinstal.

- [Penawaran pasar dengan driver 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. Untuk informasi selengkapnya, lihat [Buat AMI dengan Windows Sysprep](#).

Opsi 2: Driver NVIDIA publik

Opsi yang ditawarkan AWS datang dengan lisensi yang diperlukan untuk pengemudi. Alternatifnya, Anda dapat menginstal driver publik dan membawa lisensi Anda sendiri. Untuk menginstal driver publik, unduh dari situs NVIDIA seperti yang dijelaskan di sini.

Atau, Anda dapat menggunakan opsi yang ditawarkan oleh AWS alih-alih driver publik. Untuk menggunakan driver GRID pada instance P3, gunakan AWS Marketplace AMI seperti yang dijelaskan dalam [Opsi 1](#). Untuk menggunakan driver GRID pada instance G6, Gr6, G5, G4dn, atau G3, gunakan AWS Marketplace AMI seperti yang dijelaskan dalam Opsi 1 atau instal driver NVIDIA yang disediakan oleh seperti yang dijelaskan dalam. AWS [Opsi 3: driver GRID \(instans G6, Gr6, G5, G4dn, dan G3\)](#)

Untuk mengunduh driver NVIDIA publik

Masuk ke instans Anda dan unduh driver NVIDIA 64-bit yang sesuai untuk jenis instans dari <http://www.nvidia.com/Download/Find.aspx>. Untuk Tipe Produk, Seri Produk, dan Produk, gunakan opsi di tabel berikut.

Instans	Tipe Produk	Seri Produk	Produk
G3	Tesla	Kelas M	M60
G4dn	Tesla	T-Series	T4
G5 ¹	Tesla	A-Series	A10
G5g ²	Tesla	T-Series	NVIDIA T4G
G6 ³	Tesla	Seri-L	L4
Gr6 ³	Tesla	Seri-L	L4
P2	Tesla	K-Series	K80
P3	Tesla	V-Series	V100
P4d	Tesla	A-Series	A100
P4de	Tesla	A-Series	A100
P5 ⁴	Tesla	H-Series	H100

¹ Instans G5 memerlukan driver versi 470.00 atau setelahnya

¹ Instans G5 memerlukan driver versi 470.82.01 atau setelahnya. Sistem operasinya adalah Linux aarch64

³ Instans G6 dan Gr6 memerlukan driver versi 525.0 atau yang lebih baru.

⁴ instance P5 memerlukan driver versi 530 atau yang lebih baru.

Untuk menginstal driver NVIDIA pada sistem operasi Linux, lihat [Panduan Mulai Cepat Instalasi Driver NVIDIA](#).

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](#).

Opsi 3: driver GRID (instans G6, Gr6, G5, G4dn, dan G3)

Unduhan ini hanya tersedia untuk AWS pelanggan. Dengan mengunduh, untuk mematuhi persyaratan AWS solusi sebagaimana dimaksud dalam Perjanjian Lisensi Pengguna Akhir Cloud NVIDIA GRID (EULA), Anda setuju untuk menggunakan perangkat lunak yang diunduh hanya untuk mengembangkan AMI untuk digunakan dengan perangkat keras NVIDIA L4, NVIDIA A10G, NVIDIA Tesla T4, atau NVIDIA Tesla M60. 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 [Dokumentasi Perangkat Lunak GPU Virtual \(vGPU\) NVIDIA®](#) di situs web NVIDIA.

Pertimbangan

- Instans G6 dan Gr6 memerlukan GRID 17 atau yang lebih baru.
- Instans G5 memerlukan GRID 13.1 atau setelahnya (atau GRID 12.4 atau setelahnya).
- Instans G3 memerlukan resolusi DNS AWS yang disediakan agar lisensi GRID berfungsi.
- [IMDSv2](#) hanya didukung dengan driver NVIDIA versi 14.0 atau lebih tinggi.
- Untuk instance Windows, jika Anda meluncurkan instans Anda 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 dengan 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 instance G3.

Amazon Linux dan Amazon Linux 2

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Hubungkan dengan instans Linux Anda.
2. Instal instans AWS CLI Linux Anda dan konfigurasi kredensi 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 kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install gcc make
```

4. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

5. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

6. Hubungkan kembali ke instans Anda setelah boot ulang.
7. Instal kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y gcc kernel-devel-$(uname -r)
```

8. 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/
```

9. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

Note

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).

11. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ 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, kunjungi [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) 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 layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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 berbasis GPU Amazon EC2](#).

CentOS 7 dan Red Hat Enterprise Linux 7

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y gcc 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, kunjungi [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 layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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 berbasis GPU Amazon EC2](#).
 - c. Instal paket desktop/workstation 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. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y make gcc 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, kunjungi [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 layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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 berbasis GPU Amazon EC2](#).
 - 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. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y make gcc 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, kunjungi [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.
- Untuk membantu memanfaatkan empat layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE DCV](#).
 - 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 berbasis GPU Amazon EC2](#).

Ubuntu dan Debian

Untuk menginstal driver NVIDIA GRID pada instans Anda

- Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.
- Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
$ sudo apt-get update -y
```

- (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
```

- Boot ulang untuk memuat versi kernel terbaru.

```
$ sudo reboot
```

5. Hubungkan kembali ke instans Anda setelah boot ulang.
6. Instal kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo apt-get install -y gcc make 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 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, kunjungi [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 layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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 berbasis GPU Amazon EC2](#).

c. Instal paket desktop/workstation GUI.

```
$ sudo apt-get install -y lightdm ubuntu-desktop
```

Sistem operasi Windows

Untuk menginstal driver NVIDIA GRID pada instans Windows Anda

1. Connect ke instance Windows Anda dan buka PowerShell jendela.
2. Konfigurasi kredensial default untuk instans AWS Tools for Windows PowerShell pada Windows Anda. Untuk informasi selengkapnya, lihat [Memulai AWS Tools for Windows PowerShell](#) di Panduan Pengguna AWS Tools for Windows PowerShell

Important

Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Unduh driver dan [Perjanjian Lisensi Pengguna Akhir Cloud NVIDIA GRID](#) dari Amazon S3 ke desktop Anda menggunakan perintah berikut PowerShell .

```
$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 [Dokumentasi Perangkat Lunak GPU Virtual \(vGPU\) NVIDIA®](#) 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).

4. Arahkan ke desktop dan klik dua kali 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.
5. (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. AWS Tools for PowerShell Di AWS Windows AMI default ke versi 32-bit dan perintah ini gagal. Sebagai gantinya, gunakan versi 64-bit yang 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-bit PowerShell.

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" /v
NvCplDisableManageLicensePage /t REG_DWORD /d 1
```

6. (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 layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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 berbasis GPU Amazon EC2](#).

Opsi 4: Driver game NVIDIA (instans 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 NVIDIA A10G, dan NVIDIA Tesla T4. Setelah menginstal perangkat lunak, Anda terikat oleh persyaratan [Perjanjian Lisensi Pengguna Akhir Cloud NVIDIA GRID](#).

Pertimbangan

- Instans G3 memerlukan resolusi DNS AWS yang disediakan agar lisensi GRID berfungsi.
- [IMDSv2](#) hanya didukung dengan driver NVIDIA versi 495.x atau lebih tinggi.

Amazon Linux dan Amazon Linux 2

Untuk menginstal driver game NVIDIA pada instans Anda

1. Hubungkan dengan instans Linux Anda.
2. Instal instans AWS CLI Linux Anda dan konfigurasi kredensi 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 kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS](#)

[terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install gcc make
```

4. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

5. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

6. Hubungkan kembali ke instans Anda setelah boot ulang.

7. Instal kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y gcc kernel-devel-$(uname -r)
```

8. 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/
```

9. Ekstrak utilitas instalasi driver game dari .zip arsip yang diunduh.

```
[ec2-user ~]$ unzip latest-driver-name.zip -d nvidia-drivers
```

10. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

11. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

Note

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).

- Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

- Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.

- Untuk versi 460.39 atau setelahnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_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"
```

- 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, kunjungi [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
```

15. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

16. (Opsional) Untuk membantu memanfaatkan tampilan tunggal dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y unzip gcc 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 vGPUW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

9. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

10. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/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 460.39 atau setelahnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_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, kunjungi [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 tampilan tunggal dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y unzip gcc 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 vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/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 460.39 atau setelahnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_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, kunjungi [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 tampilan tunggal dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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.
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 kompiler gcc dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y unzip gcc make 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 vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/Linux/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 460.39 atau setelahnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_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, kunjungi [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 tampilan tunggal dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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.
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 kompiler `gcc` dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo apt-get install -y unzip gcc make 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 vGPUSW-*vGaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

10. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
$ chmod +x nvidia-drivers/Linux/NVIDIA-Linux-x86_64*-grid.run
```

11. Jalankan penginstal menggunakan perintah berikut:

```
$ sudo ./nvidia-drivers/Linux/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 460.39 atau setelahnya:

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_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, kunjungi [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 tampilan tunggal dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE DCV](#). Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah ini.

Sistem operasi Windows

Sebelum Anda menginstal driver game NVIDIA pada instans Anda, Anda harus memastikan bahwa prasyarat berikut terpenuhi selain pertimbangan yang disebutkan untuk semua driver game.

- Jika Anda meluncurkan instans Windows menggunakan AMI Windows kustom, AMI harus berupa gambar standar yang dibuat dengan Windows Sysprep untuk memastikan bahwa driver game berfungsi. Untuk informasi selengkapnya, lihat [Buat AMI dengan Windows Sysprep](#).
- Konfigurasi kredensial default untuk instans AWS Tools for Windows PowerShell pada Windows Anda. Untuk informasi selengkapnya, lihat [Memulai AWS Tools for Windows PowerShell](#) di Panduan Pengguna AWS Tools for Windows PowerShell
- Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Untuk menginstal driver game NVIDIA pada instans Windows Anda

1. Connect ke instance Windows Anda dan buka PowerShell jendela.
2. Unduh dan instal driver game menggunakan PowerShell perintah berikut.

```
$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 variabel `$KeyPrefix` dari "windows/latest" menjadi "windows".

3. Arahkan ke desktop dan klik dua kali 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: vGamingMarketplace

tipe: DWord

nilai: 2

PowerShell

Jalankan PowerShell perintah berikut untuk membuat nilai registri ini. AWS Tools for PowerShell Di AWS Windows AMI 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: vGamingMarketplace

tipe: DWord

nilai: 2

PowerShell

Jalankan PowerShell perintah berikut untuk membuat nilai registri ini. AWS Tools for PowerShell Di AWS Windows AMI 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 461.40 atau setelahnya:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-  
Archive/GridSwCertWindows_2023_9_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"
```

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
```

6. Booting ulang instans Anda.

7. Verifikasi lisensi NVIDIA Gaming menggunakan perintah berikut.

```
C:\Windows\System32\DriverStore\FileRepository\nv_dispswi.inf_*\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 dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE DCV](#). Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah ini.

Menginstal CUDA versi tambahan

Setelah Anda menginstal driver grafik NVIDIA pada instans, Anda dapat menginstal versi CUDA selain versi yang disertakan dengan driver grafik tersebut. Prosedur berikut menunjukkan cara mengonfigurasi banyak versi CUDA pada instans.

Instal toolkit CUDA di Linux

Ikuti langkah-langkah ini untuk menginstal toolkit CUDA di Linux:

1. Hubungkan dengan instans Linux Anda.
2. Buka [Situs web NVIDIA](#) dan pilih versi CUDA yang Anda butuhkan.
3. Pilih arsitektur, distribusi, dan versi untuk sistem operasi pada instans Anda. Untuk Tipe Penginstal, pilih runfile (lokal).
4. Ikuti petunjuk untuk mengunduh skrip penginstalan.
5. Tambahkan izin proses ke skrip penginstalan yang Anda unduh menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x downloaded_installer_file
```

6. Jalankan skrip instal sebagai berikut untuk menginstal kit alat CUDA dan menambahkan nomor versi CUDA ke jalur kit alat.

```
[ec2-user ~]$ sudo sh downloaded_installer_file --silent --override --toolkit --samples --toolkitpath=/usr/local/cuda-version --samplespath=/usr/local/cuda --no-opengl-libs
```

7. (Opsional) Atur versi CUDA default sebagai berikut.

```
[ec2-user ~]$ sudo ln -s /usr/local/cuda-version /usr/local/cuda
```

Instal toolkit CUDA di Windows

Ikuti langkah-langkah ini untuk menginstal toolkit CUDA di Windows:

Untuk memasang kit alat CUDA

1. Hubungkan ke instans Windows Anda.
2. Buka [Situs web NVIDIA](#) dan pilih versi CUDA yang Anda butuhkan.
3. Untuk Tipe Penginstal, pilih exe (lokal), lalu pilih Unduh.
4. Menggunakan browser Anda, jalankan file instal yang diunduh. Ikuti petunjuk untuk menginstal kit alat CUDA. Anda mungkin diminta melakukan boot ulang instans.

Instal driver AMD di instans Amazon 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 [Menginstal driver NVIDIA](#) sebagai gantinya.

Daftar Isi

- [Driver Perangkat Lunak AMD Radeon Pro untuk Korporasi](#)
- [AMI dengan driver AMD terinstal](#)
- [Mengunduh driver AMD](#)
- [Siapkan desktop interaktif untuk Linux](#)

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 berbagai Amazon Machine Images (AMI) 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 4.14

Note

AMD driver versi amdgpu-pro-20.20-1184451 dan rilis driver yang lebih baru memerlukan kernel versi 5.15 atau lebih tinggi.

- Windows Server 2016
- Windows Server 2019

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 persyaratan [Perjanjian Lisensi Pengguna Akhir AMD Software](#).

Instal driver AMD pada instance Linux Anda

1. Hubungkan dengan instans Linux Anda.
2. Instal instans AWS CLI Linux Anda dan konfigurasi kredensi 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 kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Install gcc dan make, jika belum terinstal.

```
$ sudo yum install gcc make
```

4. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

- Untuk Amazon Linux 2:

```
$ sudo amazon-linux-extras install epel -y  
$ sudo yum update -y
```

- Untuk Ubuntu 22.04:

```
$ wget https://repo.radeon.com/.preview/a0e4ef1dffbc95b4abb54e891f265e61/amdgpu-  
install/5.5.02.05.2/ubuntu/jammy/amdgpu-install_5.5.02.05.50502-1_all.deb  
$ sudo apt install ./amdgpu-install_5.5.02.05.50502-1_all.deb  
$ sudo sed -i 's#repo.radeon.com#&/.preview/a0e4ef1dffbc95b4abb54e891f265e61#' /  
etc/apt/sources.list.d/{amdgpu.list,rocm.list,amdgpu-proprietary.list}
```

- Untuk versi Ubuntu lainnya:

```
$ sudo dpkg --add-architecture i386  
$ sudo apt-get update -y && sudo apt upgrade -y
```

- Untuk CentOS:

```
$ sudo yum install epel-release -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.

 Note

Lewati langkah ini untuk Ubuntu 22.04.

```
$ aws s3 cp --recursive s3://ec2-amd-linux-drivers/latest/ .
```

8. Ekstrak file.

- Untuk Amazon Linux 2 dan CentOS:

```
$ tar -xf amdgpu-pro-*rhel*.tar.xz
```

- Untuk Ubuntu:

 Note

Lewati langkah ini untuk Ubuntu 22.04.

```
$ tar -xf amdgpu-pro*ubuntu*.xz
```

9. Ubah ke folder untuk driver yang diekstrak.

10. Tambahkan modul yang hilang untuk penginstalan driver.

- Untuk Amazon Linux 2 dan CentOS:

Lewatkan langkah ini.

- Untuk Ubuntu:

 Note

Lewati langkah ini untuk Ubuntu 22.04.

```
$ sudo apt install linux-modules-extra-$(uname -r) -y
```

11. Jalankan skrip menginstal mandiri untuk menginstal tumpukan grafis penuh.

- Untuk Ubuntu 22.04:

```
$ sudo amdgpu-install --usecase=workstation --vulkan=pro --opengl=rocr,legacy -y
```

- Untuk Amazon Linux 2 dan CentOS dan versi Ubuntu lainnya:

```
$ ./amdgpu-pro-install -y --opengl=pal,legacy
```

12. Boot ulang instans.

```
$ sudo reboot
```

13. Konfirmasikan bahwa pengemudi berfungsi.

```
$ dmesg | grep amdgpu
```

Responsnya akan terlihat seperti berikut:

```
Initialized amdgpu
```

Instal driver AMD pada instance Windows Anda

1. Connect ke instance Windows Anda dan buka PowerShell jendela.

2. Konfigurasi kredensial default untuk instans AWS Tools for Windows PowerShell pada Windows Anda. Untuk informasi selengkapnya, lihat [Memulai AWS Tools for Windows PowerShell](#) di Panduan Pengguna AWS Tools for Windows PowerShell

 **Important**

Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi kebijakan Akses AmazonS3 ReadOnly. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola: ReadOnly Akses AmazonS3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Unduh driver dari Amazon S3 ke desktop Anda menggunakan perintah berikut PowerShell .

```
$Bucket = "ec2-amd-windows-drivers"
$KeyPrefix = "latest" # use "archives" for Windows Server 2016
$LocalPath = "$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
    }
}
```

4. Buka zip file driver yang diunduh dan jalankan penginstal menggunakan perintah berikut PowerShell .

```
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
```

- Ikuti petunjuk untuk menginstal driver dan mem-boot ulang instans Anda sesuai kebutuhan.
- Untuk memverifikasi bahwa GPU berfungsi dengan benar, periksa Device Manager. Anda akan melihat “AMD Radeon Pro V520 MxGPU” terdaftar sebagai adaptor tampilan.
- Untuk membantu memanfaatkan empat layar dengan resolusi hingga 4K, siapkan protokol tampilan performa tinggi [NICE 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.

```
$ C:\> sudo amazon-linux-extras install epel -y
```

2. Instal desktop MATE.

```
$ C:\> sudo amazon-linux-extras install mate-desktop1.x -y
$ C:\> sudo yum groupinstall "MATE Desktop" -y
$ C:\> sudo systemctl disable firewalld
```

3. Salin file `xorg.conf` ke `/etc/X11/xorg.conf`.
4. Boot ulang instans.

```
$ C:\> sudo reboot
```

5. (Opsional) [Instal server NICE DCV](#) untuk menggunakan NICE DCV sebagai protokol tampilan performa tinggi, dan kemudian [menghubungkan ke sesi NICE DCV](#) menggunakan klien pilihan Anda.

Untuk menyiapkan desktop interaktif di Ubuntu

1. Instal desktop MATE.

```
$ sudo apt install xorg-dev ubuntu-mate-desktop -y
$ C:\> 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 NICE DCV](#) untuk menggunakan NICE DCV sebagai protokol tampilan performa tinggi, dan kemudian [menghubungkan ke sesi NICE DCV](#) menggunakan klien pilihan Anda.
6. Setelah penginstalan DCV memberikan izin video Pengguna DCV:

```
$ sudo usermod -aG video dcw
```

Untuk menyiapkan desktop interaktif di CentOS

1. Instal repositori EPEL.

```
$ sudo yum update -y
$ C:\> sudo yum install epel-release -y
```

2. Instal desktop MATE.

```
$ sudo yum groupinstall "MATE Desktop" -y
$ C:\> sudo systemctl disable firewalld
```

3. Salin file `xorg.conf` ke `/etc/X11/xorg.conf`.
4. Boot ulang instans.

```
$ sudo reboot
```

5. (Opsional) [Instal server NICE DCV](#) untuk menggunakan NICE DCV sebagai protokol tampilan performa tinggi, dan kemudian [menghubungkan ke sesi NICE DCV](#) menggunakan klien pilihan Anda.

Driver paravirtual untuk instans Windows

AMI Windows berisi seperangkat driver untuk memungkinkan 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.

	RedHat PV	Citrix PV	AWS PV
Jenis instans	Tidak didukung untuk semua tipe instans. Jika Anda menentukan tipe instans yang tidak didukung, instans	Didukung untuk tipe instans Xen.	Didukung untuk tipe instans Xen.

	RedHat PV	Citrix PV	AWS PV
	tersebut akan mengalami gangguan.		
Volume terlampir	Mendukung hingga 16 volume terlampir.	Mendukung lebih dari 16 volume terlampir.	Mendukung lebih dari 16 volume terlampir.

	RedHat 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. l. Ketika instance berada dalam grup penempatan cluster, ini menawarkan kinerja jaringan yang lebih baik antara instance yang ada di grup penempatan cluster. Untuk informasi selengkapnya, lihat Grup penempatan .

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 2022	AWS PV versi terbaru
Windows Server 2019	AWS PV versi terbaru
Windows Server 2016	AWS PV versi terbaru
Windows Server 2012 R2	AWS PV versi terbaru
Windows Server 2012	AWS PV versi terbaru
Windows Server 2008 R2	AWS PV versi 8.3.5
Windows Server 2008	Citrix PV 5.9
Windows Server 2003	Citrix PV 5.9

Daftar Isi

- [AWS Driver PV](#)
- [Driver Citrix PV](#)
- [RedHat Driver PV](#)
- [Berlangganan notifikasi](#)
- [Mutakhirkan driver PV pada instans Windows](#)
- [Memecahkan masalah driver PV pada instance Windows](#)

AWS Driver PV

Driver AWS PV disimpan di %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
```

Komponen driver AWS PV tercantum dalam 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 instance generasi Xen. Anda dapat mengakses dan mengelola layanan dengan menjalankan `Services.msc` dari baris perintah. Saat berjalan pada instance generasi Nitro, driver AWS PV tidak digunakan dan `LiteAgent` layanan akan berhenti sendiri dimulai dengan driver versi 8.2.4. Memperbarui ke driver AWS PV terbaru juga memperbarui `LiteAgent` dan meningkatkan keandalan pada semua generasi instans.

Instal driver AWS PV terbaru

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

- Anda dapat menggunakan AWS Systems Manager untuk memperbarui driver PV secara otomatis. Untuk informasi selengkapnya, lihat [Panduan: Secara Otomatis Perbarui Driver PV pada Instans EC2 Windows \(Konsol\)](#) di Panduan Pengguna AWS Systems Manager .
- Anda dapat [mengunduh](#) paket driver dan menjalankan program instalasi secara manual. Pastikan untuk memeriksa file `readme.txt` untuk mengetahui persyaratan sistem. Untuk informasi tentang mengunduh dan menginstal driver AWS PV, atau memutakhirkan kontroler domain, lihat [Tingkatkan instans Windows Server \(peningkatan AWS PV\) secara manual](#).

AWS Riwayat paket driver PV

Tabel berikut menunjukkan perubahan driver AWS PV untuk setiap rilis driver.

Versi paket	Detail	Tanggal rilis
8.4.3	Memperbaiki bug di penginstal paket untuk meningkatkan pengalaman pemutakhiran.	24 Januari 2023
8.4.2	Perbaiki stabilitas untuk mengatasi kondisi balapan.	13 April 2022
8.4.1	Penginstal paket yang ditingkatkan.	7 Januari 2022
8.4.0	<ul style="list-style-type: none"> Perbaiki stabilitas untuk mengatasi kasus langka terjebak disk IO. Perbaiki stabilitas untuk mengatasi kasus crash yang jarang terjadi selama pelepasan volume EBS. Menambahkan fitur untuk mendistribusikan beban di banyak inti untuk beban kerja yang memanfaatkan lebih dari 20.000 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. AWS Instalasi PV 8.4 pada Windows Server 2008 R2 akan gagal. AWS PV versi 8.3.5 dan sebelumnya didukung pada Windows Server 2008 R2. 	2 Maret 2021
8.3.5	Penginstal paket yang ditingkatkan.	7 Januari 2022
8.3.4	Peningkatan keandalan lampiran perangkat jaringan.	4 Agustus 2020
8.3.3	<ul style="list-style-type: none"> Perbarui ke komponen yang XenStore menghadap ke -facing untuk mencegah pemeriksaan bug selama jalur penanganan kesalahan. Perbarui ke komponen penyimpanan untuk menghindari kerusakan ketika SRB tidak valid dikirimkan. 	4 Februari 2020

Versi paket	Detail	Tanggal rilis
	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 .	
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.	15 Januari 2019
8.2.5	Peningkatan keamanan tambahan. 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. 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. Untuk memverifikasi bahwa versi ini telah diinstal, lihat nilai registri Windows berikut ini: HKLM\Software\Amazon\PVDriver\Version 8.2.1 .	8 Maret 2018
7.4.6	Perbaikan stabilitas untuk membuat driver AWS PV lebih tangguh.	26 April 2017

Versi paket	Detail	Tanggal rilis
7.4.3	<p>Dukungan tambahan untuk Windows Server 2016.</p> <p>Perbaiki stabilitas untuk semua versi OS Windows yang didukung.</p> <p>* Tanda tangan driver AWS PV versi 7.4.3 berakhir pada 29 Maret 2019. Kami merekomendasikan memperbarui ke driver AWS PV terbaru.</p>	18 November 2016
7.4.2	Perbaiki stabilitas untuk dukungan tipe instans X1.	2 Agu 2016
7.4.1	<ul style="list-style-type: none"> • Peningkatan kinerja pada driver AWS PV Storage. • Perbaiki stabilitas di driver AWS PV Storage: Memperbaiki masalah saat instance mengalami crash sistem dengan kode pemeriksaan bug 0x0000Dead. • Perbaiki stabilitas pada driver Jaringan AWS PV. • Menambahkan dukungan untuk Windows Server 2008R2. 	12 Juli 2016
7.3.2	<ul style="list-style-type: none"> • Peningkatan logging dan diagnostik. • Perbaiki stabilitas pada driver AWS PV Storage. Dalam beberapa kasus, disk mungkin tidak muncul di Windows setelah menyambungkan kembali disk ke instans. • Menambahkan dukungan untuk Windows Server 2012. 	24 Juni 2015
7.3.1	<p>Pembaruan TRIM: Perbaiki terkait dengan permintaan TRIM. Perbaikan ini menstabilkan instans dan meningkatkan kinerja instans saat mengelola permintaan TRIM dalam jumlah besar.</p>	
7.3.0	<p>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 penyimpanan berbasis EBS tidak mendukung TRIM mulai Maret 2015.</p>	

Versi paket	Detail	Tanggal rilis
7.2.5	<ul style="list-style-type: none">• Perbaiki stabilitas pada driver AWS PV Storage: Dalam beberapa kasus driver AWS PV dapat menurunkan memori yang tidak valid dan menyebabkan kegagalan sistem.• Perbaiki stabilitas saat menghasilkan crash dump: Dalam beberapa kasus pengemudi AWS 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	<p>Persistensi ID Perangkat: Perbaiki 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 ini juga mencakup modifikasi pada co-installer untuk driver AWS PV sehingga sistem tetap dipetakan perangkat virtual.</p>	
7.2.2	<ul style="list-style-type: none">• Muat driver AWS PV dalam mode Directory Services Restore Mode (DSRM): Directory Services Restore Mode adalah opsi boot mode aman untuk pengontrol domain Windows Server.• 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.	

Versi paket	Detail	Tanggal rilis
7.2.1	<ul style="list-style-type: none"> Jalankan di mode aman: Memperbaiki masalah di mana driver tidak mau memuat dalam mode aman. Sebelumnya Driver AWS PV hanya akan membuat instance dalam sistem yang berjalan normal. 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 kluster karena disk PV bukan disk kluster yang valid. 	
7.2.0	Basis: Versi dasar AWS PV.	

Driver Citrix PV

Driver Citrix PV disimpan di direktori `%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](#).

RedHat Driver PV

RedHat driver didukung untuk instans lama, tetapi tidak direkomendasikan pada instans yang lebih baru dengan RAM lebih dari 12GB karena keterbatasan driver. Instans dengan lebih dari 12GB RAM yang menjalankan RedHat driver dapat gagal untuk boot dan menjadi tidak dapat diakses.

Kami merekomendasikan untuk meningkatkan RedHat driver ke driver Citrix PV, dan kemudian meningkatkan driver Citrix PV ke driver PV. AWS

File sumber untuk RedHat driver ada di direktori %ProgramFiles%\RedHat (instance 32-bit) atau %ProgramFiles(x86)%\RedHat (instance 64-bit). Kedua driver tersebut adalah `rhe1net`, driver jaringan RedHat Paravirtualized `rhe1scsi`, dan driver miniport SCSI. RedHat

Berlangganan notifikasi

Amazon SNS dapat memberi Anda notifikasi saat EC2 Windows Drivers versi baru dirilis. Gunakan salah satu metode berikut untuk berlangganan notifikasi ini.

Note

Anda harus menentukan Wilayah untuk Topik SNS langganan Anda.

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.

Berlangganan notifikasi EC2 menggunakan AWS CLI

Untuk berlangganan pemberitahuan EC2 dengan AWS CLI, gunakan perintah berikut.

```
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
```

Berlangganan notifikasi EC2 menggunakan AWS Tools for PowerShell

Untuk berlangganan pemberitahuan EC2 dengan Tools untuk Windows PowerShell, gunakan perintah berikut.

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-  
drivers' -Region us-east-1 -Protocol email -Endpoint 'YourUserName@YourDomainName.ext'
```

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 Langganan.
3. Pilih kotak centang untuk berlangganan lalu pilih Tindakan, Hapus berlangganan. Ketika diminta untuk mengonfirmasi, pilih Hapus.

Mutakhirkan driver PV pada instans Windows

Kami menyarankan Anda menginstal driver PV terbaru untuk meningkatkan stabilitas dan kinerja 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 Koneksi Jaringan di Panel Kontrol dan lihat Koneksi Area Lokal. Periksa apakah driver tersebut adalah salah satu dari yang berikut:

- AWS Perangkat Jaringan PV
- Adaptor Ethernet Citrix PV
- RedHat Pengemudi PV NIC

Atau, Anda dapat memeriksa output dari perintah `pnputil -e`.

Persyaratan sistem

Pastikan untuk memeriksa file `readme.txt` di unduhan untuk mengetahui persyaratan sistem.

Daftar Isi

- [Tingkatkan instans Windows Server \(peningkatan AWS PV\) dengan Distributor](#)
- [Tingkatkan instans Windows Server \(peningkatan AWS PV\) secara manual](#)
- [Tingkatkan pengontrol domain \(peningkatan AWS PV\)](#)
- [Mutakhirkan instans Windows Server 2008 dan 2008 R2 \(pemutakhiran Redhat ke Citrix PV\)](#)
- [Mutakhirkan layanan agen tamu Citrix Xen Anda](#)

Tingkatkan instans Windows Server (peningkatan AWS PV) dengan Distributor

Anda dapat menggunakan Distributor, kemampuan AWS Systems Manager, untuk menginstal atau meng-upgrade paket driver AWS 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 \(peningkatan AWS PV\)](#). Proses pemutakhiran versi untuk instans kontroler domain berbeda dari edisi standar Windows.

1. Kami menyarankan Anda membuat cadangan jika Anda perlu memutar kembali perubahan Anda.

Tip

Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Systems Manager Automation untuk membuat AMI menggunakan runbook. `AWS-CreateImage` Untuk informasi selengkapnya, lihat [AWS-CreateImage](#) di Panduan Pengguna referensi buku 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 “Connect ke instans Windows Anda 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 cara menginstal atau meng-upgrade 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. Konfigurasi parameter lain untuk paket seperlunya dan jalankan instalasi atau tingkatkan menggunakan prosedur yang direferensikan di [Step 4](#).

Setelah menjalankan paket Distributor, instance secara otomatis 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 telah 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 apa pun dan membawanya secara online sesuai dengan huruf drive yang tercantum dalam [Step 3](#).

Jika sebelumnya Anda dinonaktifkan [Pemindahan TCP](#) menggunakan Netsh untuk driver PV Citrix, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke driver PV. AWS Masalah pembongkaran TCP dengan driver Citrix tidak ada di driver PV. AWS Hasilnya, 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 AWS driver PV.

Tingkatkan instans Windows Server (peningkatan AWS PV) secara manual

Gunakan prosedur berikut untuk melakukan pemutakhiran driver AWS PV di tempat, atau untuk meningkatkan 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 RedHat driver, atau untuk versi Windows Server lainnya.

Beberapa versi Windows Server sebelumnya tidak dapat menggunakan driver terbaru. Untuk memverifikasi versi driver mana yang akan digunakan untuk sistem operasi Anda, lihat tabel versi driver di halaman [Driver paravirtual untuk instans Windows](#).

 Important

Jika instans Anda adalah kontroler domain, lihat [Tingkatkan pengontrol domain \(peningkatan AWS PV\)](#). Proses pemutakhiran versi untuk instans kontroler domain berbeda dari edisi standar Windows.

Untuk meningkatkan driver AWS PV secara manual

1. Kami menyarankan Anda membuat cadangan jika Anda perlu memutar kembali perubahan Anda.

i Tip

Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Systems Manager Automation untuk membuat AMI menggunakan runbook. `AWS-CreateImage` Untuk informasi selengkapnya, lihat [AWS-CreateImage](#) di Panduan Pengguna referensi buku 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](#) paket driver terbaru ke instans.

Atau, jalankan PowerShell 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
```

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
```

5. Ekstrak konten folder, lalu 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 telah diinstal dengan menghubungkan ke instance menggunakan Remote Desktop dan kemudian menjalankan perintah berikut: PowerShell

```
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 apa pun dan membawanya secara online sesuai dengan huruf drive yang tercantum dalam [Step 3](#).

Jika sebelumnya Anda dinonaktifkan [Pemindahan TCP](#) menggunakan Netsh untuk driver PV Citrix, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke driver PV. AWS Masalah pembongkaran TCP dengan driver Citrix tidak ada di driver PV. AWS Hasilnya, 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 AWS driver PV.

Tingkatkan pengontrol domain (peningkatan AWS PV)

Gunakan prosedur berikut pada pengontrol domain untuk melakukan peningkatan driver AWS PV di tempat, atau untuk meningkatkan dari driver Citrix PV ke AWS driver PV.

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 untuk Kontroler Domain Virtual](#) di 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 PV. AWS 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 instans.
5. Ekstrak konten folder, lalu 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 instans dengan menentukan nama pengguna dalam format berikut `hostname\administrator`. Misalnya, `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 dinonaktifkan [Pemindahan TCP](#) menggunakan Netsh untuk driver PV Citrix, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke Driver PV. AWS Masalah pembongkaran TCP dengan driver Citrix tidak ada di driver PV. AWS Hasilnya, TCP Offloading memberikan kinerja yang lebih baik dengan driver AWS PV.

Mutakhirkan instans Windows Server 2008 dan 2008 R2 (pemutakhiran Redhat ke Citrix PV)

Sebelum Anda mulai meningkatkan RedHat driver Anda ke driver Citrix PV, pastikan Anda melakukan hal berikut:

- Instal versi terbaru dari layanan EC2Config. Untuk informasi selengkapnya, lihat [Menginstal EC2Config versi terbaru](#).
- Verifikasi bahwa Anda telah menginstal Windows PowerShell 3.0. Untuk memverifikasi versi yang telah Anda instal, jalankan perintah berikut di 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 Microsoft Download Center.

- Cadangkan informasi penting Anda pada instans, atau buat AMI dari instans. Untuk informasi selengkapnya tentang cara membuat AMI, lihat [Buat AMI yang didukung Amazon EBS-Backed](#).

Tip

Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Systems Manager Automation untuk membuat AMI menggunakan runbook. [AWS-CreateImage](#) Untuk informasi selengkapnya, lihat [AWS-CreateImage](#) di Panduan Pengguna referensi buku 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 informasi selengkapnya, lihat [Mengkonfigurasi Pengaturan TCP/IP](#) di Perpustakaan Microsoft. TechNet

Untuk meng-upgrade RedHat driver

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal. Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat [Hubungkan ke instans Windows Anda](#).
2. Dalam instans Anda, [unduh](#) paket pemutakhiran Citrix PV.
3. Ekstrak konten paket yang dimutakhirkan ke lokasi pilihan Anda.
4. Klik dua kali 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 untuk Windows uninstaller, pilih Ya untuk menghapus perangkat lunak. RedHat 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>
L79ThJPF8LyIL38I2ht0FBrjet3vnT2csTiU/XGVMRCH7kQtBznAnXrKd1sirXlx19BwVMsd9b38jFJqv01IUpgNNJRZoCdc7IbUw
</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: U
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.

```

PVUpgrade - Notepad
File Edit Format View Help
20130315_0905:25 Reinstall Device PCI\IDE\IDECHANNEL\4&80001ED6060
20130315_0905:33 Reinstall Device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0609
20130315_0905:43 Reinstall Device ACPI\PNP0A03\0
20130315_0905:49 Removing Service: rhelftr
20130315_0905:49 Removing Service: rhelnet
20130315_0905:49 Removing Service: rhelscs1
20130315_0905:49 Removing Driver File: C:\Windows\System32\drivers\rhelftr.sys
20130315_0905:50 Removing Driver File: C:\Windows\System32\drivers\rhelnet.sys
20130315_0905:50 Removing Redhat Service: C:\Windows\System32\rhelsvc.exe
20130315_0905:50 unable to delete file, need to restart
20130315_0905:50 Restarting computer...
20130315_0905:50 -----
20130315_0907:05 START: 20130315_0907
20130315_0907:05 Running as: SYSTEM
20130315_0907:05 Current Running Directory: C:\Users\Administrator\downloads\Citrix-Win_PV
20130315_0907:05 Detecting Windows version
20130315_0907:16 Reinstall Device PCI\IDE\IDECHANNEL\4&80001ED6060
20130315_0907:43 Reinstall Device PCI\IDE\IDECHANNEL\4&80001ED6060
20130315_0907:49 Reinstall Device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0609
20130315_0907:57 Reinstall Device ACPI\PNP0A03\0
20130315_0908:05 Removing Redhat Service: C:\Windows\System32\rhelsvc.exe
20130315_0908:05 Removing Driver File: C:\Windows\System32\drivers\rhelscs1.sys
20130315_0908:08 Adding quick Removal Settings to: C:\Windows\System32\DriverStore\FileRepository\disk_inf_126712d3\disk_inf
20130315_0908:08 Adding first Surprise Removal Item
20130315_0908:08 Adding last surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing disks for quick Removal
20130315_0908:08 complete
20130315_0908:08 Adding quick Removal Settings to: C:\Windows\System32\DriverStore\FileRepository\disk_inf_5c850fad\disk_inf
20130315_0908:08 Adding first Surprise Removal Item
20130315_0908:08 Adding last surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing disks for quick Removal
20130315_0908:08 complete
20130315_0908:08 Removing Scheduled Task
20130315_0908:09
20130315_0908:09 IMPORTANT: Please uninstall any remaining Redhat driver software from Add/Remove Programs
20130315_0908:09
20130315_0908:09 INSTALLATION IS COMPLETE
20130315_0908:09 Setting Powershell script execution policy to Restricted

```

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 meningkatkan ke driver AWS 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 AMI yang didukung Amazon EBS-Backed](#).

Tip

Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Systems Manager Automation untuk membuat AMI menggunakan runbook. `AWS-CreateImage` Untuk informasi selengkapnya, lihat [AWS-CreateImage](#) di Panduan Pengguna referensi buku 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](#).
2. Pada instans, [unduh](#) paket pemutakhiran Citrix.
3. Ekstrak konten paket yang dimutakhirkan ke lokasi pilihan Anda.

4. Klik dua kali 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 instance 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 AWS Management Console 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 memerlukan tipe instans C3. Jika server saat ini tidak menggunakan 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

Metode	Prasyarat	Ringkasan Prosedur
		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, melampirkannya 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 menyelesaikan masalah.

Metode 1 - Jaringan yang ditingkatkan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Contoh.
3. Cari instans yang terpengaruh. Pilih instans dan pilih status Instans, lalu pilih Hentikan instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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. Connect 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 Contoh.
3. Cari instans yang terpengaruh. Pilih instans, pilih status Instans, lalu pilih Hentikan instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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.

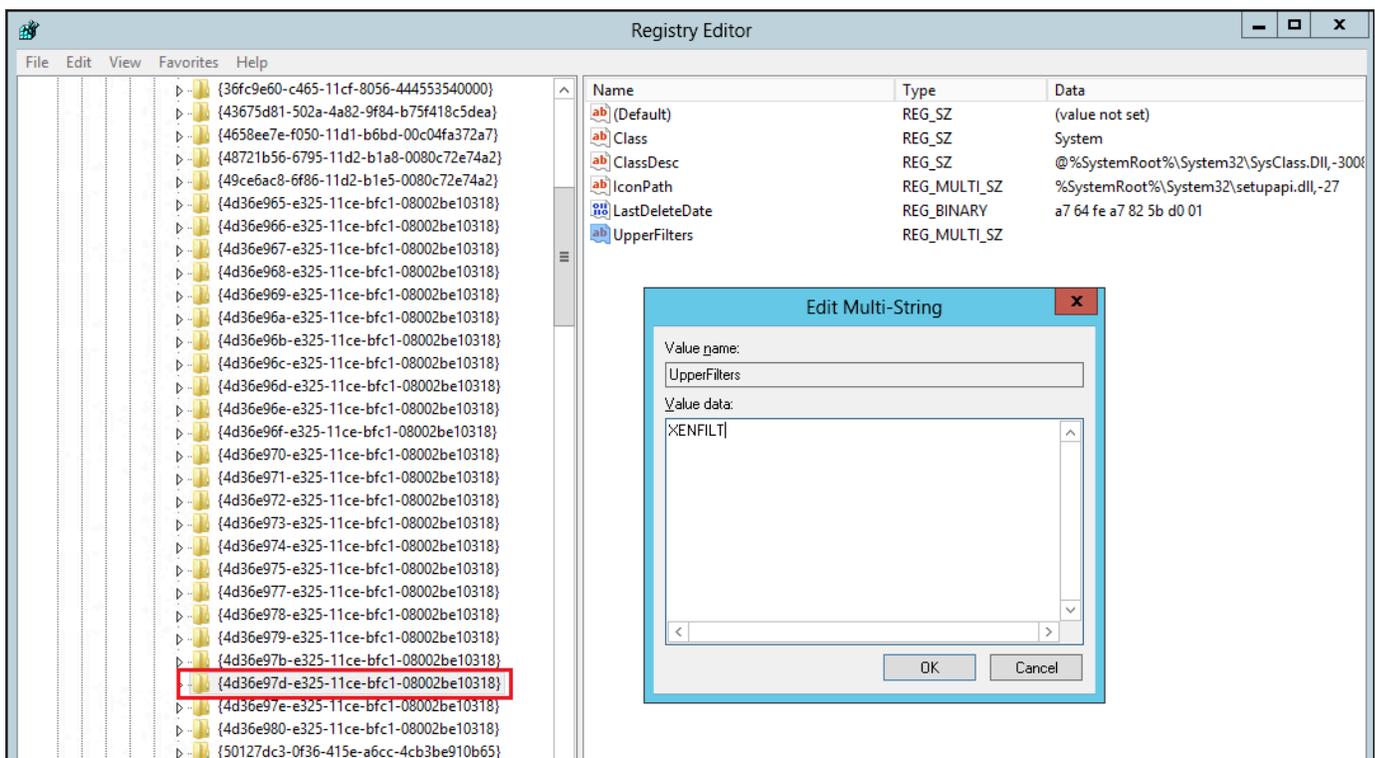
5. Pada panel navigasi, pilih Volume.

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 Editor Registri, pilih HKEY_Local_Machine, lalu dari menu File pilih Muat 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*your_temporary_key_name*\ 001\ Kontrol\ Kelas\
4d36e97d-e325-11ce-bfc1-08002be10318 ControlSet

HKEY_LOCAL_MACHINE*your_temporary_key_name*\ 001\ Kontrol\ Kelas\
4d36e96a-e325-11ce-bfc1-08002be10318 ControlSet

12. Untuk setiap tombol, klik dua kali UpperFilters, masukkan nilai XENFILT, lalu pilih OK.



13. Temukan kunci berikut:

HKEY_LOCAL_MACHINE*your_temporary_key_name*\ 001\ Layanan\ XENBUS\
Parameter ControlSet

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*your_temporary_key_name*\ 001\ Layanan\ XENBUS ControlSet

16. Ubah Count dari 0 menjadi 1.

17. Temukan dan hapus kunci berikut:

HKEY_LOCAL_MACHINE*your_temporary_key_name*\ 001\ Layanan\ xenvbd\
ControlSet StartOverride

HKEY_LOCAL_MACHINE*your_temporary_key_name*\ 001\ Layanan\ xenfilt\
ControlSet 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. Connect 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 Contoh.
3. Pilih instans yang ingin Anda jalankan skrip remediasinya. Pilih Status instans, lalu pilih Mulai instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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. Connect ke instance dengan menggunakan Remote Desktop dan kemudian [download RemediateDriverIssue folder.zip](#) ke instance.
7. Ekstrak isi folder tersebut.
8. Jalankan skrip remediasi sesuai petunjuk di file Readme.txt. File tersebut terletak di folder tempat Anda RemediateDriverIssue mengekstraksi.zip.

Pemindahan TCP

⚠ Important

Masalah ini tidak berlaku untuk instance yang menjalankan driver jaringan AWS PV atau Intel.

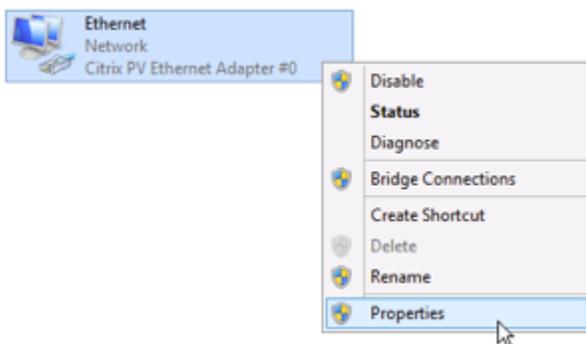
Secara default, pemindahan TCP diaktifkan untuk driver Citrix PV di AMI Windows. Jika Anda mengalami kesalahan tingkat pengangkutan atau kesalahan transmisi paket (seperti yang terlihat di Windows Performance Monitor)—misalnya, saat Anda menjalankan beban kerja SQL tertentu—Anda mungkin perlu menonaktifkan 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. Klik kanan 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 Lanjutan, nonaktifkan setiap properti, kecuali untuk Nilai Checksum TCP/UDP 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`.

Pembacaan dan penulisan disk (iOS) dalam 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 tahap persiapan atau penyelesaian menjadi hambatan tergantung pada sifat-sifat iOS yang digunakan oleh aplikasi.

Dimulai dengan driver AWS PV 8.4.0, beban fase persiapan dan fase penyelesaian dapat didistribusikan di beberapa core, menghilangkan kemacetan. 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 [Mutakhirkan driver PV pada instans Windows](#).
- Anda memiliki akses RDP ke instans. Untuk langkah-langkah agar ter-connect ke instans Windows menggunakan RDP, lihat [Connect ke instans 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. Klik kanan grafik pada panel utama dan pilih Ubah grafik ke>Prosesor logis untuk menampilkan masing-masing 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 tersisir.
- 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 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 direkomenasikan untuk semua instans Xen menggunakan PV 8.4.0 atau setelahnya dan memanfaatkan lebih dari 20.000 IOPS, baik ditemukan masalah ataupun tidak.
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 versi driver AWS 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

Atur kunci registri `NotifierDistributed` untuk memungkinkan driver penyimpanan PV untuk memilih apakah akan mendistribusikan penyelesaian IO atau tidak.

1. Hubungkan ke instans Anda menggunakan RDP.
2. Buka PowerShell command prompt baru sebagai administrator.
3. Jalankan perintah berikut untuk mengatur kunci registri `NotifierDistributed`.

```
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.

AWS Driver NVMe untuk instance Windows

Volume Amazon EBS dan volume penyimpanan instans diekspos sebagai perangkat pemblokiran NVMe pada [instans yang dibangun di](#) Sistem Nitro. AWS Untuk sepenuhnya memanfaatkan kinerja dan kemampuan fitur Amazon EBS untuk volume yang diekspos sebagai perangkat blok NVMe, instans harus menginstal driver NVMe. AWS Semua AMI AWS Windows generasi saat ini hadir 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](#).

Instal atau tingkatkan driver AWS NVMe menggunakan PowerShell

Jika Anda tidak menggunakan AMI AWS Windows terbaru yang disediakan oleh Amazon, gunakan prosedur berikut untuk menginstal driver AWS NVMe saat ini. Anda harus melakukan pembaruan ini pada saat yang tepat untuk melakukan boot ulang instans Anda. Entah skrip instalasi akan mem-boot ulang instans Anda atau Anda harus mem-boot ulang sebagai langkah terakhir.

Prasyarat

PowerShell 3.0 atau yang lebih baru

Untuk mengunduh dan menginstal driver AWS NVMe terbaru

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 dan ekstrak driver ke instans Anda menggunakan salah satu opsi berikut:
 - Menggunakan peramban:
 - a. [Unduh](#) paket driver terbaru ke instans.
 - b. Ekstrak arsip zip.
 - Menggunakan PowerShell:

```
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
```

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. 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 NVMe1.5.0, reservasi persisten Antarmuka Sistem Komputer Kecil (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 NVMe1.5.0, `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.

Instal atau tingkatkan driver AWS NVMe dengan Distributor

Anda dapat menggunakan Distributor, kemampuan AWS Systems Manager, untuk menginstal paket driver NVMe satu kali atau dengan pembaruan terjadwal.

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 Nama, pilih AWSNVMe.
3. Untuk jenis instalasi, pilih Uninstall dan instal ulang.
4. (Opsional) Sesuaikan instalasi dengan menentukan nilai untuk `AdditionalArguments`.
 - a. Dimulai dengan AWS NVMe1.5.0, 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}` untuk `AdditionalArguments`. Jika Anda tidak ingin mengaktifkan fitur ini, tentukan `{"SSM_EnableSCSIPersistentReservations": $false}` untuk `AdditionalArguments`.
 - b. Dimulai dengan AWS NVMe1.5.0, `install.ps1` skrip akan selalu menginstal alat `ebsnvme-id`

(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 `{"SSM_InstallEBSNVMeIdTool": "Yes"}` untuk `AdditionalArguments`.
- ii. Jika Anda tidak ingin menginstal alat, tentukan `{"SSM_InstallEBSNVMeIdTool": "No"}` untuk `AdditionalArguments`.

Jika `SSM_InstallEBSNVMeIdTool` tidak ditentukan untuk `AdditionalArguments`, 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. Argumen tambahan harus diformat menggunakan sintaks JSON yang valid. Untuk contoh cara meneruskan argumen tambahan untuk paket `aws configure`, lihat [dokumentasi Amazon EC2 Systems Manager](#).

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.

Konfigurasi reservasi persisten SCSI

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 NVMe

Tabel berikut menjelaskan versi driver AWS NVMe yang dirilis.

Versi paket	Versi Driver	Detail	Tanggal rilis
1.5.1	1.5.0	Memperbaiki skrip instalasi untuk membuat folder untuk alat <code>ebsnvme-id</code> jika tidak ada.	17 November 2023
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 <code>ebsnvme-id</code> (<code>ebsnvme-id.exe</code>) sekarang diinstal secara default.	31 Agustus 2023

Versi paket	Versi Driver	Detail	Tanggal rilis
1.4.2	1.4.2	Memperbaiki bug yang Driver AWS NVMe tidak mendukung volume penyimpanan instans pada instans 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 selengkapnya, lihat bagian 8.25, "Meningkatkan Performa melalui Kepatuhan Ukuran I/O dan Keselrasan," di Spesifikasi Dasar NVMe, versi 1.4 .	20 Mei 2022
1.4.0	1.4.0	<ul style="list-style-type: none"> Menambahkan dukungan untuk IOCTL yang memungkinkan aplikasi berinteraksi dengan perangkat NVMe. Dukungan ini memungkinkan aplikasi untuk mendapatkan daftar <code>IdentifyController</code>, <code>IdentifyNamespace</code>, dan <code>Namespace</code> dari perangkat NVMe. Untuk informasi selengkapnya, lihat Kueri spesifik protokol di dokumentasi Microsoft. AWSNVMe Instalasi 1.4.0 pada Windows Server 2008 R2 akan gagal. AWSNVMe versi 1.3.2 dan sebelumnya didukung pada Windows Server 2008 R2. Versi driver 1.4.0 dan alat <code>ebsnvme-id</code> terbaru (<code>ebsnvme-id.exe</code>) digabungkan dalam satu paket. Kombinasi ini memungkinkan Anda untuk menginstal driver dan alat dari satu paket. Untuk detail selengkapnya, lihat Instal atau tingkatkan driver AWS NVMe menggunakan PowerShell. Perbaiki bug dan peningkatan keandalan. 	23 November 2021

Versi paket	Versi Driver	Detail	Tanggal rilis
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.	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 di semua instans yang didukung, termasuk instans bare metal.	13 Juni 2018
1.0.0	1.0.0	AWS Driver NVMe untuk jenis instans yang didukung yang 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 Langganan.
3. Pilih kotak centang untuk berlangganan lalu pilih Tindakan, Hapus berlangganan. Ketika diminta untuk mengonfirmasi, pilih Hapus.

Untuk berlangganan pemberitahuan EC2 menggunakan AWS CLI

Untuk berlangganan pemberitahuan 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 pemberitahuan 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'
```

Konfigurasi instans Windows Anda

Setelah meluncurkan instance Windows, Anda dapat masuk sebagai administrator untuk melakukan konfigurasi tambahan untuk agen peluncuran dan fitur khusus Windows. Topik berikut fokus pada konfigurasi instance Windows.

Daftar Isi

- [Konfigurasi pengaturan peluncuran untuk instans Windows Amazon EC2](#)
- [Gunakan EC2 Fast Launch untuk instans Windows Anda](#)
- [Gunakan akselerator Amazon Elastic Graphics pada instans Windows](#)
- [Menginstal WSL pada instans Windows Anda](#)

Konfigurasi pengaturan peluncuran untuk instans Windows Amazon EC2

Agan peluncuran Amazon EC2 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.

- [Konfigurasi instans Windows menggunakan EC2Launch v2](#)
- [Konfigurasi instans Windows menggunakan EC2Launch](#)
- [Konfigurasi instance Windows menggunakan layanan EC2config \(legacy\)](#)

Daftar isi

- [Bandingkan agen peluncuran Amazon EC2](#)
- [Konfigurasi DNS Suffix untuk agen peluncuran Windows](#)

Bandingkan agen peluncuran Amazon EC2

Tabel berikut menunjukkan perbedaan fungsional utama antara EC2Config, EC2Launch v1, dan EC2Launch v2.

Fitur	EC2Config	EC2Launch v1	EC2Launch v2
Jalankan sebagai	Layanan Windows	PowerShell Skrip	Layanan Windows
Mendukung	Hanya OS warisan	Windows 2016 Windows 2019 (LTSC dan SAC)	Windows 2016 Windows 2019 (LTSC dan SAC) Windows 2022
File konfigurasi	XML	XML	YAML
Tetapkan nama pengguna Administrator	Tidak	Tidak	Ya
Ukuran data pengguna	16 KB	16 KB	60 KB (terkompresi)
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 hanya untuk referensi historis. Versi sistem operasi yang dijalkannya tidak lagi didukung oleh Microsoft. Kami sangat menyarankan Anda meningkatkan ke layanan peluncuran terbaru.

Konfigurasi DNS Suffix untuk agen peluncuran 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 di kunci `System\CurrentControlSet\Services\Tcpip\Parameters\SearchList` registri dengan menambahkan nilai berikut ke daftar pencarian akhiran DNS:

- Domain dari 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. [setDnsSuffix](#)
- Untuk informasi tentang pengaturan daftar akhiran DNS dan cara mengaktifkan atau menonaktifkan devolusi untuk EC2launch v1, lihat. [Konfigurasi EC2Launch](#)
- 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 dalam domain anak untuk mengakses sumber daya di domain induk tanpa menggunakan nama domain yang sepenuhnya memenuhi syarat. Secara default, devolusi nama domain berlanjut hingga hanya ada dua node yang tersisa dalam perkembangan nama domain.

Agan peluncuran melakukan devolusi pada nama domain jika instance terhubung ke domain, dan menambahkan hasilnya ke daftar pencarian akhiran DNS yang dipertahankan 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**
 - Ketika tidak diatur, menonaktifkan devolusi
 - Saat disetel ke 1, aktifkan devolusi (default)
 - Ketika diatur ke 0, menonaktifkan devolusi
- **System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel**
 - Bila tidak disetel, gunakan level of 2 (default)
 - Saat disetel ke 3 atau lebih besar, gunakan nilai untuk mengatur level

Ketika Anda menonaktifkan devolusi atau mengubah pengaturan devolusi Anda ke tingkat yang lebih tinggi, kunci **System\CurrentControlSet\Services\Tcpip\Parameters\SearchList** registri stil 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 mengatur 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. Berkembang ke `corp.example.com`. Perkembangan berhenti di sini, karena pengaturan level.

Konfigurasi instans Windows menggunakan EC2Launch v2

Semua instans Amazon EC2 yang didukung yang menjalankan Windows Server 2022 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-*`.

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`). Biner layanan terletak di direktori `%ProgramFiles%\Amazon\EC2Launch`.

EC2Launch v2 menerbitkan log peristiwa Windows untuk membantu Anda memecahkan masalah kesalahan dan mengatur pemacu. Untuk informasi selengkapnya, lihat [Log peristiwa Windows](#).

Sistem operasi yang didukung

- Windows Server 2022
- Windows Server 2019 (Saluran Layanan Jangka Panjang dan Saluran Semi-Tahunan)
- Windows Server 2016

Isi bagian EC2Launch v2

- [Gambaran umum EC2Launch v2](#)
- [Instal EC2Launch v2 versi terbaru](#)
- [Migrasikan ke EC2Launch v2](#)
- [Hentikan, mulai ulang, hapus, atau uninstal EC2Launch v2](#)
- [Berlangganan notifikasi layanan EC2Launch v2](#)
- [Pengaturan EC2Launch v2](#)
- [Penyelesaian masalah EC2Launch v2](#)
- [Riwayat versi EC2Launch v2](#)

Gambaran umum EC2Launch v2

EC2Launch v2 adalah layanan yang menjalankan tugas selama permulaan instans dan berjalan jika sebuah instans dihentikan dan kemudian dimulai, atau dimulai ulang.

Ringkasan topik

- [Konsep EC2Launch v2](#)
- [Tugas EC2Launch v2](#)
- [Telemetri](#)

Untuk membandingkan fitur versi agen peluncuran, lihat [Bandingkan agen peluncuran Amazon EC2](#).

Konsep EC2Launch v2

Konsep berikut berguna untuk dipahami saat mempertimbangkan EC2Launch v2.

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](#).

Tahap

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 PostReadytahap. Beberapa versi skrip berjalan sebelum PostReadytahap `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.

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

agent-config

`agent-config` adalah file yang terletak di folder konfigurasi untuk EC2Launch v2. Ini termasuk konfigurasi untuk boot, jaringan PreReady, dan PostReady tahapan. 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.

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 [Bagaimana Amazon EC2 menangani data pengguna untuk instans Windows](#).

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 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
- Setel partisi non-boot ke `mbx` atau `gpt`
- 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.

Telemetri

Telemetri adalah informasi tambahan yang membantu AWS untuk lebih memahami kebutuhan Anda, mendiagnosis masalah, dan memberikan fitur untuk meningkatkan pengalaman Anda. Layanan AWS

EC2Launch v2 versi 2.0.592 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, jalankan kode kesalahan, dan tumpukan panggilan 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.

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 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
```

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.
- Luncurkan instans Anda dari AMI yang telah diinstal sebelumnya EC2launch v2.

Warning

AmazonEC2Launch.msi menghapus instalasi layanan peluncuran EC2 versi sebelumnya, seperti EC2Launch (v1) dan EC2Config.

Untuk langkah penginstalan, pilih tab yang cocok dengan metode pilihan Anda.

Windows 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"
```

- 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
```

- Untuk memverifikasi instalasi, periksa apakah file msi ada di direktori EC2Launch v2 pada instans Anda (C:\ProgramData\Amazon\EC2Launch).

AWS Systems Manager Distributor

Untuk mengonfigurasi pembaruan otomatis untuk EC2Launch v2 dengan AWS Systems Manager Quick Setup, lihat [Instal dan perbarui secara otomatis dengan Pengaturan Cepat Distributor](#)

Anda juga dapat melakukan instalasi satu kali AWSEC2Launch-Agent paket 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 di AMI Windows Server 2022 dan UEFI:

- Windows_Server-2022-English-Full-Base
- Windows_Server-2022-English-Core-Base
- AMI Windows Server 2022 dengan semua bahasa lainnya
- AMI Windows Server 2022 dengan SQL diinstal
- Windows_Server-2022-English-Core-EKS_Optimized

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
- EC2LaunchV2-Windows_Server-2012_R2_RTM-English-Full-Base
- EC2LaunchV2-Windows_Server-2012_RTM-English-Full-Base

Secara otomatis menginstal dan memperbarui EC2launch v2 dengan AWS Systems Manager Distributor Quick Setup

Dengan Pengaturan Cepat AWS Systems Manager Distributor, Anda dapat mengatur pembaruan otomatis untuk EC2launch v2. Proses berikut menyiapkan Systems Manager Association pada instans Anda yang secara otomatis memperbarui agen EC2launch v2 pada frekuensi yang Anda tentukan. Asosiasi yang dibuat oleh Penyiapan Cepat Distributor dapat menyertakan instance dalam Wilayah Akun AWS dan, atau instans dalam Organisasi. AWS Untuk informasi selengkapnya tentang menyiapkan organisasi, lihat [Tutorial: Membuat dan mengonfigurasi organisasi](#) di Panduan AWS Organizations Pengguna.

Sebelum Anda mulai, pastikan bahwa contoh Anda memenuhi semua prasyarat.

Prasyarat

Untuk mengatur pembaruan otomatis dengan Penyiapan Cepat Distributor, instans Anda harus memenuhi prasyarat berikut.

- Anda memiliki setidaknya satu instance yang berjalan yang mendukung EC2launch v2. Lihat sistem operasi yang didukung untuk [EC2Launch v2](#).
- Anda telah melakukan tugas penyiapan Systems Manager pada instans Anda. Untuk informasi selengkapnya, lihat [Menyiapkan Systems Manager](#) di Panduan AWS Systems Manager Pengguna.
- EC2launch v2 harus menjadi satu-satunya agen peluncuran yang diinstal pada instans Anda. Jika Anda memiliki lebih dari satu agen peluncuran yang diinstal, konfigurasi Pengaturan Cepat Distributor Anda akan gagal. Sebelum Anda mengonfigurasi EC2launch v2 dengan Penyiapan Cepat Distributor, hapus instalasi agen peluncuran EC2config atau EC2launch v1, jika ada.

Konfigurasi Pengaturan Cepat Distributor untuk EC2launch v2

Untuk membuat konfigurasi EC2launch v2 dengan Penyiapan Cepat Distributor, gunakan pengaturan berikut saat Anda menyelesaikan langkah-langkah penerapan [paket Distributor](#):

- Paket perangkat lunak: Agen Amazon EC2launch v2.
- Frekuensi pembaruan: Pilih frekuensi dari daftar.
- Target: Pilih dari opsi penerapan yang tersedia.

Untuk memeriksa status konfigurasi Anda, navigasikan ke tab Systems Manager Quick Setup Configurations di AWS Management Console.

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 Region, status Deployment, dan status Asosiasi.

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 pemecahan masalah, lihat [Memecahkan Masalah hasil Penyiapan Cepat di Panduan Pengguna](#).AWS Systems Manager

Unduhan EC2Launch v2 di Amazon S3

Untuk menginstal EC2Launch v2 versi terbaru, unduh penginstal dari salah satu lokasi berikut:

 Note

Tautan instalasi 32-bit akan usang. Kami menyarankan Anda menggunakan tautan 64-bit untuk menginstal ke EC2Launch v2. Jika Anda memerlukan agen peluncuran 32-bit, gunakan [EC2config](#).

- 64Bit — <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/amd64/latest/AmazonEC2Launch.msi>
- 32Bit — <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/386/latest/AmazonEC2Launch.msi>

Konfigurasi opsi instalasi

Saat Anda menginstal atau memutakhirkan EC2Launch v2, Anda dapat mengonfigurasi opsi instalasi dengan dialog instalasi EC2Launch v2 atau dengan perintah msixexec 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.

Opsi Instal Dasar

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 mengunggah data. Jika tidak, itu tidak mengunggah.

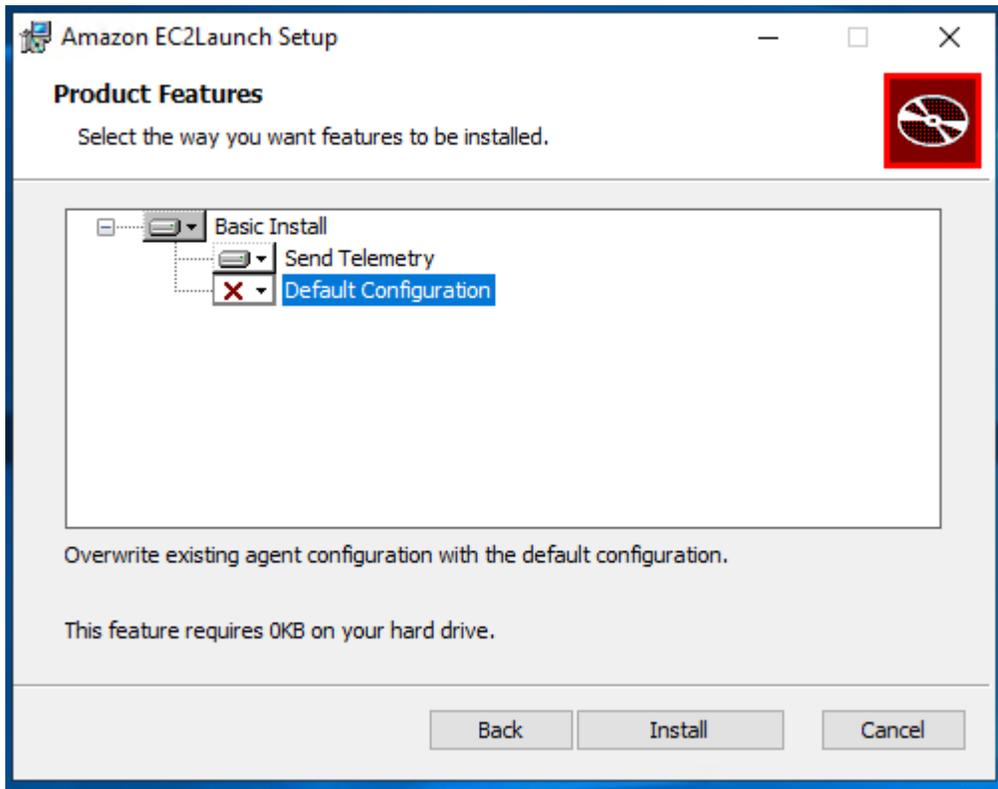
Konfigurasi default

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`, penginstal mengatur variabel lingkungan `EC2LAUNCH_TELEMETRY` ke nilai `1`.

Ketika Anda membiarkan nilai Telemetry keluar dari parameter ADDLOCAL, 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 telemetry. Jika nilainya sama dengan 1, agen mengunggah data. Jika tidak, itu tidak mengunggah.

Contoh: instal EC2Launch v2 dengan telemetry

```
& msixexec /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.

Windows 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 Windows AMI](#) di Referensi AMI AWS Windows.

Untuk EC2Launch v2 versi terbaru, lihat [Riwayat versi EC2Launch v2](#).

Untuk alat migrasi EC2Launch v2 versi terbaru, lihat [Riwayat versi alat migrasi EC2Launch v2](#).

Anda dapat menerima notifikasi saat layanan EC2Launch v2 versi baru dirilis. Untuk informasi selengkapnya, lihat [Berlangganan notifikasi layanan EC2Launch v2](#).

Migrasikan ke EC2Launch v2

Alat migrasi EC2Launch memutakhirkan agen peluncuran yang diinstal (EC2Config dan EC2Launch v1) dengan melepas instalasi dan memasang EC2Launch v2. Konfigurasi yang berlaku dari layanan peluncuran sebelumnya secara otomatis dimigrasikan ke layanan baru. 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](#).

Anda dapat mengunduh alat migrasi atau menginstal dengan RunCommand dokumen SSM.

Anda dapat mengunduh alat dari lokasi berikut:

Note

Tautan alat migrasi 32-bit tidak akan digunakan lagi. Kami menyarankan Anda menggunakan tautan 64-bit untuk bermigrasi ke EC2Launch v2. Jika Anda memerlukan agen peluncuran 32-bit, gunakan [EC2config](#).

- 64Bit - <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/Windows/AMD64/terbaru/EC2 Tool.zip LaunchMigration>
- 32Bit - <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/Windows/386/terbaru/EC2 Tool.zip LaunchMigration>

Note

Anda harus menjalankan alat migrasi EC2Launch v2 sebagai Administrator. EC2Launch v2 diinstal sebagai layanan setelah Anda menjalankan alat migrasi. Ia tidak langsung berjalan. Secara default, layanan ini melakukan tugas-tugas selama startup instans dan berjalan jika sebuah instans dihentikan dan kemudian dimulai, atau dimulai ulang.

Menggunakan dokumen SSM [AWSEC2Launch-RunMigration](#) untuk bermigrasi ke EC2Launch v2 versi terbaru dengan SSM RunCommand. Dokumen tidak membutuhkan parameter apa pun. Untuk informasi selengkapnya tentang menggunakan SSM RunCommand, lihat [AWS Run Command Systems Manager](#).

Alat migrasi menerapkan konfigurasi berikut dari EC2Config ke EC2Launch v2.

- Jika `Ec2DynamicBootVolumeSize` diatur ke, `false` hapus tahap boot EC2Launch v2
- Jika `Ec2SetPassword` diatur ke, `Enabled`atur tipe kata sandi EC2Launch v2 ke `random`
- Jika `Ec2SetPassword` diatur ke, `Disabled`atur tipe kata sandi EC2Launch v2 ke `do nothing`
- Jika `SetDnsSuffixList` diatur ke, `false` hapus tugas `setDnsSuffix` EC2Launch v2
- Jika `EC2SetComputerName` diatur ke `BETUL`, tambahkan tugas `setHostName` EC2Launch v2 ke konfigurasi `yaml`

Alat migrasi menerapkan konfigurasi berikut dari EC2Launch v1 ke EC2Launch v2.

- Jika `ExtendBootVolumeSize` diatur ke, `false` hapus tahap boot EC2Launch v2
- Jika `AdminPasswordType` diatur ke, `Random`atur tipe kata sandi EC2Launch v2 ke `random`
- Jika `AdminPasswordType` diatur ke, `Specify`atur tipe kata sandi EC2Launch v2 ke `static` dan data kata sandi ke kata sandi yang ditentukan dalam `AdminPassword`
- Jika `SetWallpaper` diatur ke, `false` hapus tugas `setWallpaper` EC2Launch v2
- Jika `AddDnsSuffixList` diatur ke, `false` hapus tugas `setDnsSuffix` EC2Launch v2
- Jika `SetComputerName` diatur ke, `true` tambahkan tugas `setHostName` EC2Launch v2

Hentikan, mulai ulang, hapus, atau uninstal EC2Launch v2

Anda dapat mengelola layanan EC2Launch v2 sama seperti yang Anda lakukan pada layanan Windows lainnya.

EC2Launch v2 berjalan satu kali saat boot dan menjalankan semua tugas yang dikonfigurasi. Setelah menjalankan tugas, layanan memasuki status berhenti. Saat Anda memulai ulang layanan, layanan akan menjalankan semua tugas yang dikonfigurasi lagi dan kembali ke status berhenti.

Untuk menerapkan pengaturan yang diperbarui ke instans, Anda dapat menghentikan dan memulai ulang layanan. Jika Anda menginstal secara manual EC2Launch v2, Anda harus menghentikan layanan terlebih dahulu.

Untuk menghentikan layanan EC2Launch v2

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, pilih Alat Administratif, lalu buka Layanan.
3. Di daftar layanan, klik kanan Amazon EC2Launch, lalu pilih Berhenti.

Untuk memulai ulang layanan EC2Launch v2

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, pilih Alat Administratif, lalu buka Layanan.
3. Di daftar layanan, klik kanan Amazon EC2Launch, lalu pilih Mulai ulang.

Jika Anda tidak perlu memperbarui pengaturan konfigurasi, membuat AMI Anda sendiri, atau menggunakan AWS Systems Manager, maka Anda dapat menghapus dan mencopot pemasangan layanan. Menghapus layanan akan menghapus subkunci registernya. Menghapus instalasi layanan akan menghapus file, subkunci registri, dan pintasan apa pun ke layanan tersebut.

Untuk menghapus layanan EC2Launch v2

1. Mulai jendela prompt perintah.
2. Jalankan perintah berikut:

```
sc delete EC2Launch
```

Untuk melepas instalasi EC2Launch v2

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, pilih Panel Kontrol.

3. Buka Program dan kemudian Program dan Fitur.
4. Di daftar program, pilih Amazon EC2Launch. Untuk mengonfirmasi bahwa Anda memilih v2, periksa kolom Versi.
5. Pilih Hapus Instalasi.

Berlangganan notifikasi layanan EC2Launch v2

Amazon SNS dapat memberi Anda notifikasi saat layanan EC2Launch v2 versi baru sedang dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Berlangganan notifikasi EC2Launch v2

1. [Masuk ke AWS Management Console dan buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.](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 ARN Topik, gunakan Amazon Resource Name (ARN) berikut: `arn:aws:sns:us-east-1:309726204594:amazon-ec2launch-v2`.
 - 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 yang meminta Anda untuk mengonfirmasi langganan Anda. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Kapan pun versi baru dari layanan EC2Launch v2 dirilis, kami akan mengirim notifikasi ke pelanggan. Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

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.

Pengaturan EC2Launch v2

Bagian ini berisi informasi tentang cara mengonfigurasi pengaturan untuk EC2Launch v2.

Topiknya mencakup:

- [Ubah pengaturan menggunakan kotak dialog pengaturan EC2Launch v2](#)
- [Struktur direktori EC2Launch v2](#)
- [Konfigurasi 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.

Amazon EC2Launch settings ✕

General | DNS suffix | Wallpaper | Volumes

Set computer name

- Set the computer name of the instance
- Set to "ip- <hex private IPv4 address> "
- Use custom name
- Reboot after setting computer name

Extend boot volume

- Extend OS partition to use free space for boot volume

Set administrator account

- Set administrator account

Administrator username (leave blank for default)

Administrator password settings

- Random (retrieve from console)
- Specify (temporarily stored in configuration file)
- Do not set

Start SSM service

- Re-enable and start SSM service after Sysprep

Optimize ENA

- Optimize receive side scaling and receive queue depth

Enable SSH

- Enable OpenSSH for later Windows versions

Enable Jumbo Frames

- Enable Jumbo Frames

Important: Do not enable Jumbo Frames if you are not familiar with them

Prepare for imaging

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. Pengaturan ini dapat berguna ketika instans di-boot dari volume perangkat 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 EC2Launch 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 Manager 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 menggunakan pada layanan EC2Launch v2 dan SSM Agent versi terbaru. Untuk informasi selengkapnya, lihat [Perbarui SSM Agent dengan menggunakan Run Command](#) dalam Panduan Pengguna AWS .

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 [Mengonfigurasi afinitas CPU RSS](#).

f. Aktifkan SSH

Pengaturan ini memungkinkan OpenSSH untuk versi Windows yang lebih baru untuk memungkinkan administrasi sistem jarak jauh.

g. Aktifkan Jumbo Frame

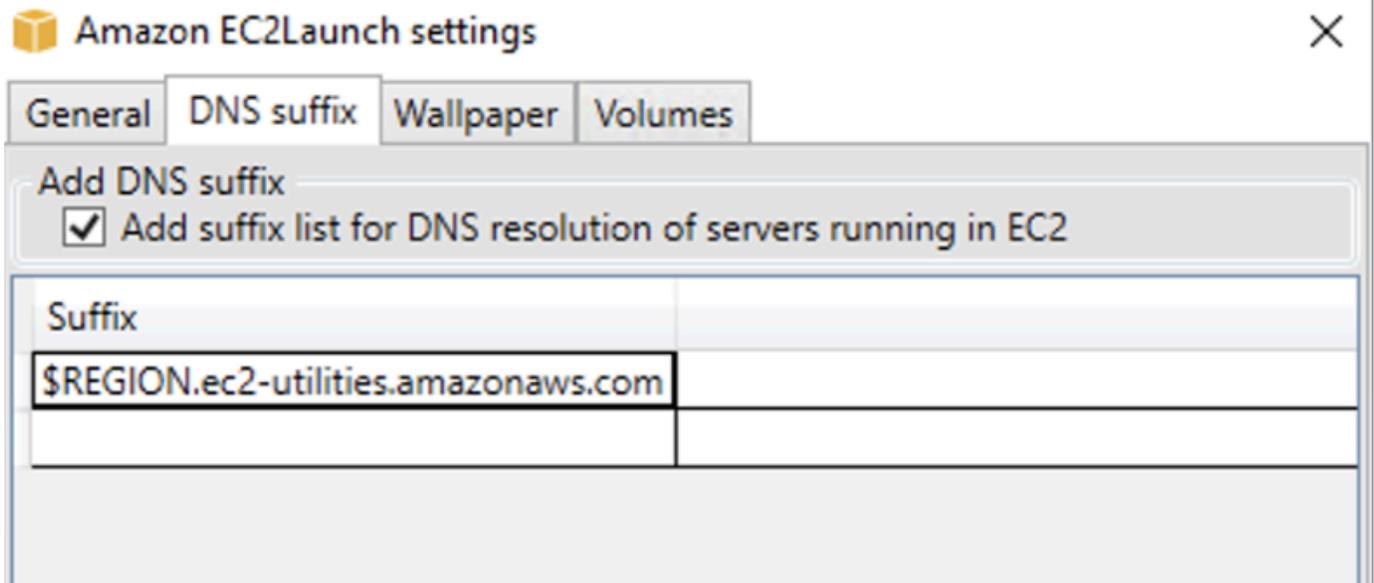
Pilih untuk mengaktifkan Jumbo Frames. Jumbo Frames dapat memiliki efek yang tidak diinginkan pada komunikasi jaringan Anda, jadi pastikan Anda memahami bagaimana Jumbo Frames akan memengaruhi sistem Anda sebelum mengaktifkan. Untuk informasi selengkapnya tentang Jumbo Frames, lihat [Frame jumbo \(9001 MTU\)](#).

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.



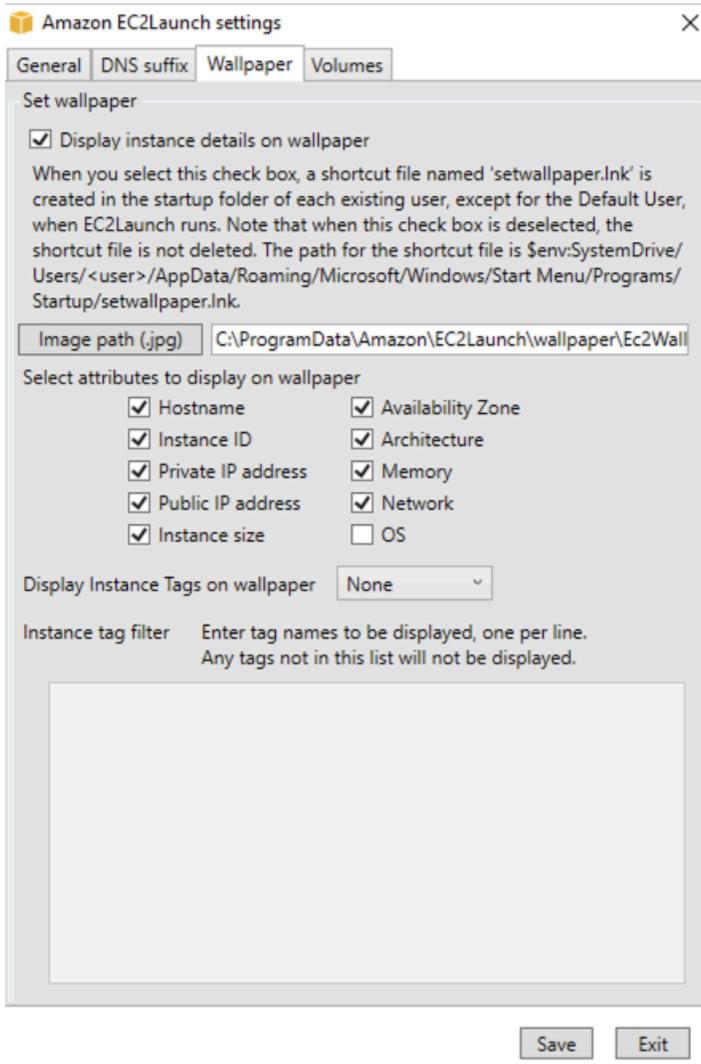
5. Pada tab Wallpaper, Anda dapat mengonfigurasi wallpaper instans Anda dengan gambar latar belakang, dan menentukan detail instans untuk wallpaper yang akan ditampilkan. Amazon EC2 menghasilkan detail setiap kali Anda masuk.

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 [Bekerja dengan tanda instans dalam 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 jenis gpt partisi, dan yang lebih kecil dari 2 TB diinisialisasi dengan jenis mbt 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.

Amazon EC2Launch settings



- General
- DNS suffix
- Wallpaper
- Volumes

Initialize volumes

Initialize All Devices

Devices

If you choose Devices, only the devices listed below are initialized. You must enter the Device for each device to be initialized. Use the devices listed on the EC2 console, for example, xvdb or /dev/nvme0n1. Name, Letter, and Partition are optional.

Device	Name	Letter	Partition
--------	------	--------	-----------

Berikut ini adalah contoh file YAML konfigurasi yang dibuat dari pengaturan yang dimasukkan di dialog EC2Launch.

```
version: 1.0
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
            - memory
            - network
  - stage: postReady
    tasks:
      - task: startSsm
```

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 file agent-config.yml
 - Yam1DotNet.dll — DLL untuk mendukung beberapa operasi di antarmuka pengguna
- tools
 - ebsnvme-id.exe — alat untuk memeriksa metadata volume EBS di instans
 - AWSAcpiSpcrReader.exe — alat untuk menentukan port COM yang benar untuk digunakan
 - EC2LaunchEventMessage.dll — DLL untuk mendukung pencatatan peristiwa Windows untuk EC2Launch.
- service
 - EC2LaunchService.exe — Layanan Windows dapat dieksekusi yang diluncurkan ketika agen peluncuran berjalan sebagai layanan.
- EC2Launch.exe — EC2Launch utama dapat dieksekusi
- EC2LaunchAgentAttribution.txt — atribusi untuk kode yang digunakan dalam EC2 Launch

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`), konsol (`console.log`), performa (`bench.log`), dan kesalahan (`error.log`) disimpan di direktori ini. File log ditambahkan ke eksekusi layanan selanjutnya.

- `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.

Konfigurasi 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.

Perintah

- [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
```

Contoh 2

PowerShell Perintah berikut menunjukkan cara mengedit dan menyimpan agent-config file dalam format JSON.

```
$config = & "$env:ProgramFiles/Amazon/EC2Launch/EC2Launch.exe" --format json |  
  ConvertFrom-Json  
$jumboFrame ="  
{  
  "task": "enableJumboFrames"  
}  
"@
```

```
$config.config | %{if($_.stage -eq 'postReady'){$_tasks += (ConvertFrom-Json -
InputObject $jumboFrame)}}
$config | ConvertTo-Json -Depth 6 | Out-File -encoding UTF8
$env:ProgramData/Amazon/EC2Launch/config/agent-config.yml
```

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
```

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,instanceSize,availabilityZone,a
```

Masukan

Parameter

--allowed-tags [**tag-name-1**, **tag-name-n**]

(Opsional) Base64 mengkode array JSON dari nama tanda instans untuk ditampilkan di wallpaper. Anda dapat menggunakan tanda ini atau --all-tags, tetapi tidak keduanya.

`--attributes` ***attribute-string-1, attribute-string-n***

(Opsional) Daftar string atribut `wallpaper` yang dipisahkan dengan koma untuk menerapkan pengaturan ke wallpaper.

`[--path | -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](#)
- [Skema: data pengguna](#)
- [Ketentuan tugas](#)

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": "1.0",
```

```
"config": [  
  {  
    "stage": "string",  
    "tasks": [  
      {  
        "task": "string",  
        "inputs": {  
          ...  
        }  
      },  
      ...  
    ]  
  },  
  ...  
]
```

YAML

```
version: 1.0  
config:  
- stage: string  
  tasks:  
  - task: string  
  inputs:  
    ...  
    ...  
    ...
```

Contoh: **agent-config.yml**

Contoh berikut menunjukkan pengaturan untuk file konfigurasi `agent-config.yml`.

```
version: 1.0  
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
        - memory
        - network
  - stage: postReady
    tasks:
      - task: startSsm
```

Skema: data pengguna

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 [Ketentuan tugas](#).

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 [Bagaimana Amazon EC2 menangani data pengguna untuk instans Windows](#).

Contoh dokumen YAMG berikut menunjukkan PowerShell skrip yang EC2launch v2 berjalan 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. Agar sesuai dengan perilaku EC2launch v1 dan EC2config, skrip data pengguna berjalan sebagai proses terlampir/inline 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	<ul style="list-style-type: none"> Tugas data pengguna berjalan sebelum tahap PostReady dalam file konfigurasi agen. Menjalankan data pengguna sebelum memulai Systems Manager Agent (perilaku yang sama seperti EC2Launch v1 dan EC2Config).* 	EC2Launch v2 versi 2.0.1245
1.0	<ul style="list-style-type: none"> Akan usang. Tugas data pengguna berjalan sebelum tahap PostReady dalam file konfigurasi agen. Ini tidak kompatibel dengan EC2Launch v1. 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.

Ketentuan tugas

Setiap tugas memiliki set properti dan persyaratan sendiri. Untuk detail, lihat tugas individual yang ingin Anda sertakan dalam dokumen.

Tugas

- [activateWindows](#)
- [enableJumboFrames](#)
- [enableOpenSsh](#)
- [executeProgram](#)
- [executeScript](#)
- [extendRootPartition](#)
- [initializeVolume](#)
- [optimizeEna](#)
- [set AdminAccount](#)
- [setDnsSuffix](#)
- [setHostName](#)
- [setWallpaper](#)
- [startSsm](#)
- [sysprep](#)
- [writeFile](#)

activateWindows

Mengaktifkan Windows terhadap satu set AWS KMS server. Aktivasi dilewati jika instans terdeteksi sebagai Bawa Lisensi Sendiri (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
```

enableJumboFrames

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
```

enableOpenSsh

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

Anda dapat mengonfigurasi parameter runtime sebagai 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 **`executeProgram`** tugas tersebut

`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)

Agan 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

Agan 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

Anda dapat mengonfigurasi parameter runtime sebagai 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. Parameter ini tidak didukung untuk tipe: 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 **executeScript** tugas tersebut

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” di 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
  content: |
    Get-Process | Out-File -FilePath C:\Process.txt
  runAs: localSystem
- frequency: always
  type: batch
  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
}
```

extendRootPartition

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 melewati inisialisasi jika mendeteksi bahwa volume tidak kosong. Volume dianggap kosong jika 4 KiB pertama dari volume adalah kosong, atau jika volume tidak memiliki [tata letak hard disk yang dapat dikenali Windows](#).

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 `mbt`.
- 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 instans Amazon 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
```

setDnsSuffix

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 [Konfigurasi DNS Suffix untuk agen peluncuran 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
```

setHostName

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 hanya berisi 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:

atribut (daftar string)

(Opsional) Anda dapat menambahkan satu atau lebih atribut berikut ke wallpaper Anda:

- `architecture`
- `availabilityZone`
- `hostName`

- `instanceId`
- `instanceSize`
- `memory`
- `network`
- `privateIpAddress`
- `publicIpAddress`

`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. Misalnya:

```
instanceTags:  
  - Tag 1  
  - Tag 2
```

`jalur (string)`

(Wajib) Jalur nama file dari file gambar format `.jpg` lokal yang akan digunakan untuk gambar wallpaper Anda.

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 [Bekerja dengan tanda instans dalam 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
```

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 XML yang 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 AMI yang didukung Amazon EBS-Backed](#).

Penyelesaian masalah 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](#)
- [Tidak valid agent-config.yml mencegah pembukaan kotak dialog pengaturan EC2Launch v2](#)
- [task:executeScript should be unique and only invoked once](#)

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 dari volume adalah kosong, atau jika volume tidak memiliki [tata letak hard disk yang dapat dikenali Windows](#). Volume yang diinisialisasi dan diformat pada sistem Linux tidak memiliki tata letak hard disk yang dikenali Windows, 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

Task `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. 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.

Tidak valid **agent-config.yml** 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 should be unique and only invoked once

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 pengenalan 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

Peristiwa terkait layanan yang dihasilkan ketika layanan dimulai atau berhenti termasuk satu digit pengenalan peristiwa.

Peristiwa	Pengenalan digit tunggal
Success	0
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 pengenalan 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 - Pesan tingkat layanan
	1 - Boot
	2 - Jaringan
	3 - PreReady
	5 - Windows sudah Siap
	6 - PostReady
	7 - Data Pengguna

Tipe peristiwa	Definisi
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 1 - Informasi 2 - Peringatan 3 - Kesalahan

Contoh ID Peristiwa

Berikut adalah contoh ID peristiwa.

- 5000 - Windows siap digunakan
- 3010- Aktifkan tugas windows di PreReady panggung berhasil
- 6013- Mengatur tugas wallpaper di tahap Data PostReady Lokal mengalami kesalahan

Skema log peristiwa Windows

MessageId/ Id Acara	Pesan peristiwa
. . .0	Success
. . .1	Informational
. . .2	Warning
. . .3	Error
x	EC2Launch service-level logs

MessageId/ Id Acara	Pesan peristiwa
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
3000	PreReady
3010	PreReady - activate_windows
3020	PreReady - install_egpu_manager
3030	PreReady - set_monitor_on

MessageId/ Id Acara	Pesan peristiwa
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
6050/7050	PostReadyLocal/UserData - execute_script
6060/7060	PostReadyLocal/UserData - manage_package

MessageId/ Id Acara	Pesan peristiwa
6070/7070	PostReadyLocal/UserData - initialize_volume
6080/7080	PostReadyLocal/UserData - write_file
6090/7090	PostReadyLocal/UserData - start_ssm
7100	PostReadyUserData - enable_op en_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 keluaran konsol instance dan cara mengaksesnya, lihat [the 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.

```
2023/11/30 20:18:53Z: Windows sysprep configuration complete.
2023/11/30 20:18:57Z: Message: Waiting for access to metadata...
2023/11/30 20:18:57Z: Message: Meta-data is now available.
2023/11/30 20:18:57Z: AMI Origin Version: 2023.11.15
2023/11/30 20:18:57Z: AMI Origin Name: Windows_Server-2022-English-Full-Base
2023/11/30 20:18:58Z: OS: Microsoft Windows NT 10.0.20348
2023/11/30 20:18:58Z: OsVersion: 10.0
```

```
2023/11/30 20:18:58Z: OsProductName: Windows Server 2022 Datacenter
2023/11/30 20:18:58Z: OsBuildLabEx: 20348.1.amd64fre.fe_release.210507-1500
2023/11/30 20:18:58Z: OsCurrentBuild: 20348
2023/11/30 20:18:58Z: OsReleaseId: 2009
2023/11/30 20:18:58Z: Language: en-US
2023/11/30 20:18:58Z: TimeZone: UTC
2023/11/30 20:18:58Z: Offset: UTC +0000
2023/11/30 20:18:58Z: Launch: EC2 Launch v2.0.1643
2023/11/30 20:18:58Z: AMI-ID: ami-1234567890abcdef1
2023/11/30 20:18:58Z: Instance-ID: i-1234567890abcdef0
2023/11/30 20:18:58Z: Instance Type: c5.large
2023/11/30 20:19:00Z: Driver: AWS NVMe Driver v1.5.0.33
2023/11/30 20:19:00Z: SubComponent: AWS NVMe Driver v1.5.0.33;
  EnableSCSIPersistentReservations: 0
2023/11/30 20:19:00Z: Driver: AWS PV Driver Package v8.4.3
2023/11/30 20:19:01Z: Driver: Amazon Elastic Network Adapter v2.6.0.0
2023/11/30 20:19:01Z: RDPCERTIFICATE-SUBJECTNAME: EC2AMAZ-S01T009
2023/11/30 20:19:01Z: RDPCERTIFICATE-THUMBPRINT:
  1234567890ABCDEF1234567890ABCDEF1234567890
2023/11/30 20:19:09Z: SSM: Amazon SSM Agent v3.2.1705.0
2023/11/30 20:19:13Z: Username: Administrator
2023/11/30 20:19:13Z: Password: <Password>
1234567890abcdef1EXAMPLEPASSWORD
</Password>
2023/11/30 20:19:14Z: User data format: no_user_data
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsTelemetryEnabled=true
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentOsArch=windows_amd64
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentCommandErrorCode=0
2023/11/30 20:19:14Z: Message: Windows is Ready to use
```

Pesan log konsol EC2Launch v2

Berikut ini adalah daftar semua pesan log konsol EC2Launch v2.

```
Message: Error EC2Launch service is stopping. {error message}
  Error setting up EC2Launch agent folders
  See instance logs for detail
  Error stopping service
  Error initializing service
Message: Windows sysprep configuration complete
Message: Invalid administrator username: {invalid username}
Message: Invalid administrator password
Username: {username}
```

```
Password: <Password>{encrypted password}</Password>
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}
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}
Microsoft-Hyper-V is installed.
Unable to get service status for vmms
Microsoft-Hyper-V is {status}
SSM: Amazon SSM Agent {version}
AWS VSS Version: {version}
Message: Windows sysprep configuration complete
Message: Windows is being configured. SysprepState is {state}
Windows is still being configured. SysprepState is {state}
Message: Windows is Ready to use
Message: Waiting for meta-data accessibility...
Message: Meta-data is now available.
Message: Still waiting for meta-data accessibility...
Message: Failed to find primary network interface...retrying...
User data format: {format}
```

Riwayat versi EC2Launch v2

Riwayat versi

- [Riwayat versi EC2Launch v2](#)
- [Riwayat versi alat migrasi EC2Launch v2](#)

Riwayat versi EC2Launch v2

Tabel berikut menjelaskan versi EC2Launch v2 yang dirilis.

Versi	Detail	Tanggal rilis
2.0.1924	<ul style="list-style-type: none"> • Memperbarui UI Pengaturan EC2launch. • Memperbarui perintah CLI wallpaper. • Memperbarui penginstal EC2launch. 	Juni 10, 2024
2.0.1914	<ul style="list-style-type: none"> • Tambahkan rute dengan alamat gateway yang tidak ditentukan (<code>0.0.0.0</code> untuk IPv4 atau <code>::</code> untuk IPv6). • Selalu tambahkan rute IPv4 dan IPv6. • Memperbaiki masalah saat nama Administrator pengguna ditambahkan ke <code>agent-config.yml</code> file saat tidak ditentukan. • Izin EC2launch v2 yang dimodifikasi. 	Juni 5, 2024
2.0.1881	<ul style="list-style-type: none"> • Menambahkan opsi kata sandi terenkripsi ke <code>setAdminAccount</code> tugas. • Menambahkan perintah CLI untuk mengenkripsi kata sandi statis di <code>agent-config.yml</code>. • Memperbaiki masalah di mana data pengguna XML tidak menambahkan PowerShell argumen saat dijalankan dengan izin Administrator. Untuk detail selengkapnya, lihat Bagaimana Amazon EC2 menangani data pengguna untuk instans Windows. • PowerShell Argumen yang disesuaikan untuk <code>executeScript</code> tugas dan skrip data pengguna saat dijalankan 	8 Mei 2024

Versi	Detail	Tanggal rilis
	<p>dengan <code>LocalSystem</code> izin. Ketika argumen kosong, agen menggunakan nilai default berikut: <code>-ExecutionPolicy Unrestricted</code> .</p> <ul style="list-style-type: none">• Mencegah pencetakan versi driver duplikat ke log konsol.	
2.0.1815	<ul style="list-style-type: none">• Penanganan kesalahan yang disesuaikan agar gagal pada masalah penyiapan kritis sebelum <code>sysprep</code>.• 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.• Tugas wallpaper dan nama host diubah untuk mendapatkan IP pribadi dari IMDS terlebih dahulu, kemudian gagal kembali ke WMI jika IMDS dinonaktifkan.• Memperbaiki masalah dengan <code>initializeVolume</code> tugas di mana <code>sc1</code> volume gagal diinisialisasi karena kesalahan sementara.	Maret 6, 2024
2.0.1739	<ul style="list-style-type: none">• Memperbaiki masalah yang mencegah kode keluar ditangkap oleh <code>executeScript</code> tugas yang dijalankan sebagai pengguna Administrator Windows.	Januari 17, 2024

Versi	Detail	Tanggal rilis
2.0.1702	<ul style="list-style-type: none">• Membatasi izin <code>Telemetry.log</code> menjadi <code>read-execute</code> saja untuk pengguna standar.• Mengonfigurasi layanan <code>EC2Launch Windows</code> untuk memulai kembali pada kegagalan start-up.• Membuat kegagalan <code>add-routes</code> dapat ditindaklanjuti dengan melakukan logging output <code>route.exe stderr</code>.• Memperbaiki masalah yang terjadi saat metrik rute berada di luar jangkauan <code>[1, 9999]</code>.• Menambahkan dukungan wallpaper ke beberapa tipe instans baru.• Memperbaiki masalah yang disebabkan oleh skrip data pengguna yang berjalan sebagai pengguna Administrator Windows dan mengirim output ke <code>stderr</code>.	4 Januari 2024

Versi	Detail	Tanggal rilis
2.0.1643	<ul style="list-style-type: none">• Memperbarui alat <code>ebsnvme-id.exe</code> ke versi 1.1.0.7.• Memperbaiki masalah dengan menerima penskalaan sisi (RSS) dan menerima pengaturan kedalaman antrean pada tipe instans logam yang dimulai dengan 'logam-*', seperti logam-48x1.• Peristiwa telemetri yang dihapus yang melaporkan perintah data pengguna XML yang memblokir agen.• Tugas <code>setDnsSuffix</code> yang diperbarui untuk membatasi devolusi nama domain berdasarkan entri registri: <code>HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel</code> .• Menambahkan tugas publik dan CLI yang menambahkan rute jaringan.• Catatan — Ini adalah versi terakhir yang secara resmi mendukung Windows Server 2012.• Catatan — Ini adalah versi terakhir yang secara resmi mendukung sistem operasi 32-bit.	4 Oktober 2023
2.0.1580	<ul style="list-style-type: none">• Cara agen peluncuran menangani kesalahan saat Anda mengubah izin file log diubah.• 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	<ul style="list-style-type: none">• Bendera <code>-block</code> usang dari <code>EC2Launch.exe</code> reset dan perintah <code>sysprep</code>.• <code>EC2Launch.exe</code> yang diperbarui untuk mendeteksi dan menangani perintah reset dan <code>sysprep</code> yang digunakan dalam tugas <code>executeScript inline</code>. Perintah tersebut menyebabkan agen berhenti berjalan setelah <code>executeScript</code> tugas menjalankannya.• Skrip data pengguna XML yang diperbarui untuk menjalankan <code>inline</code> secara default.• Aktifkan skrip data pengguna XML untuk berjalan terpisah dengan tanda baru <code>detach</code>. Untuk detail selengkapnya, lihat Skrip data pengguna.• Membuat perubahan berikut ini pada log agen.<ul style="list-style-type: none">• Pesan log agen yang diperbarui.• Menghapus <code>executeScript</code> konten dan output dari log agen.• Argumen <code>executeProgram</code> dan output yang dihapus dari log agen.• Membuat perubahan berikut pada log konsol.<ul style="list-style-type: none">• Menambahkan nilai <code>EnableSCSIPersistentReservations</code> ke log konsol.	3 Juli 2023

Versi	Detail	Tanggal rilis
2.0.1303	<ul style="list-style-type: none">• Menambahkan penanganan kesalahan tambahan dan baris log saat menambahkan rute jaringan.• Diizinkan <code>executeScript</code> dan <code>executeProgram</code> tugas di <code>PreReady</code> panggung.• Tugas <code>executeProgram</code> yang diperbarui untuk menghasilkan file output yang mirip dengan output dari tugas <code>executeScript</code>. Untuk informasi selengkapnya, lihat executeProgram.• Menambahkan telemetri untuk memantau penggunaan perintah agen pemblokiran dalam data pengguna XML.	3 Mei 2023
2.0.1245	<ul style="list-style-type: none">• Peningkatan visibilitas tentang kerusakan dengan mencatat tumpukan panggilan rusak dalam teks yang jelas.• Menambahkan <code>EventLog</code> layanan sebagai dependensi startup untuk memperbaiki kerusakan saat layanan Amazon EC2launch dimulai lebih cepat daripada layanan. <code>EventLog</code>• Membuat data pengguna XHTML berjalan sebelum <code>PostReady</code> tahap dari file konfigurasi agen (seperti <code>EC2launch v1</code> dan <code>EC2config</code>).• Menambahkan data pengguna YAMB versi 1.1 untuk membuat data pengguna berjalan sebelum <code>PostReady</code> tahap dari file konfigurasi agen (data pengguna YAMB versi 1.0 berjalan setelah <code>PostReady</code> tahap dari file konfigurasi agen).	8 Maret 2023

Versi	Detail	Tanggal rilis
2.0.1173	<ul style="list-style-type: none">• Menambahkan fitur opsional untuk menampilkan tanda instans pada wallpaper. Untuk informasi selengkapnya, lihat setWallpaper .• Menambahkan penanganan kesalahan saat grup keamanan untuk Elastic Graphics tidak disiapkan dengan benar.• Memperbaiki batas waktu saat Layanan Metadata Instans tidak diaktifkan.	6 Februari 2023
2.0.1121	<ul style="list-style-type: none">• Memperbaiki masalah saat kesalahan 404 dicetak ke wallpaper saat tidak ada alamat IPv4 publik yang ditetapkan.• Memperbaiki masalah saat sistem file volume diformat sebagai RAW ganti NTFS saat huruf drive perangkatnya disetel ke. D• Memperbaiki masalah di mana volume SSD NVMe salah diidentifikasi sebagai volume EBS.• Memperbaiki kesalahan saat mengaktifkan Windows saat IMDS dinonaktifkan.	4 Januari 2023

Versi	Detail	Tanggal rilis
2.0.1082	<ul style="list-style-type: none">• Memperbaiki masalah di mana bidang <code>setWallpaper : privateIpAddress</code> kosong saat IMDS dinonaktifkan.• Memperbaiki masalah dengan mengatur nama host ke alamat IPv4 pribadi saat IMDS dinonaktifkan.• Memperbaiki masalah dengan menginisialisasi volume pada Windows Server 2012.• Memperbaiki masalah dengan pengaturan bingkai jumbo.• Memperbaiki kesalahan ketika tidak ada kunci SSH yang ditentukan pada peluncuran instans.• Memperbaiki kesalahan pada Windows Server 2012 ketika Windows tidak memiliki <code>ReleaseId</code> 'kunci registri'.	7 Desember 2022
2.0.1011	<ul style="list-style-type: none">• Memperbaiki logika untuk menemukan adaptor jaringan ketika <code>PnPDeviceID</code> kosong.	11 November 2022
2.0.1009	<ul style="list-style-type: none">• Menggunakan informasi segmen PCI untuk memilih port konsol.	8 November 2022

Versi	Detail	Tanggal rilis
2.0.982	<ul style="list-style-type: none">• Menambahkan logika coba lagi untuk mendapatkan informasi RDP.• Memperbaiki kesalahan selama inisialisasi volume pada <code>d2.8xlarge</code> instans.• Memperbaiki masalah di mana adaptor jaringan yang salah dapat dipilih setelah reboot.• Menghapus pesan kesalahan alarm palsu saat ACPI SPCR tidak tersedia.	31 Oktober 2022
2.0.863	<ul style="list-style-type: none">• Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja.• Menambahkan logika untuk menetapkan huruf drive ke volume yang sudah diinisialisasi tetapi tidak dipasang.• Mencetak pesan kesalahan yang lebih spesifik ketika jenis key pair tidak didukung.• Memperbaiki bug kode reboot 3010.• Menambahkan pemeriksaan untuk data pengguna yang dengan encode base64 tidak valid.	6 Juli 2022
2.0.698	<ul style="list-style-type: none">• Memperbaiki kesalahan ketik dalam output log saat menjalankan skrip.	30 Januari 2022

Versi	Detail	Tanggal rilis
2.0.674	<ul style="list-style-type: none">• Telemetri mengunggah kontrol privasi yang diaktifkan/dinonaktifkan.• Memperbaiki <code>index out of bounds</code> bug.• Menghapus pintasan wallpaper selama <code>sysprep</code>.	15 November 2021
2.0.651	<ul style="list-style-type: none">• Menambahkan logika untuk menghapus instalasi agen warisan selama instalasi <code>EC2Launch v2</code>.• Memperbaiki masalah <code>list-volume</code> CLI saat volume root tidak terdaftar sebagai volume 0.	7 Oktober 2021
2.0.592	<ul style="list-style-type: none">• Memperbaiki bug untuk melaporkan status tahap dengan benar.• Menghapus pesan kesalahan alarm palsu saat file log ditutup.• Menambahkan telemetri.	31 Agustus 2021
2.0.548	<ul style="list-style-type: none">• Menambahkan angka nol di depan untuk nama host IP hex.• Memperbaiki izin file untuk <code>enableOpenSsh</code> tugas.• Memperbaiki crash perintah <code>sysprep</code>.	4 Agustus 2021

Versi	Detail	Tanggal rilis
2.0.470	<ul style="list-style-type: none">• Memperbaiki bug di tahap jaringan untuk menunggu DHCP menetapkan IP ke instans.• Memperbaiki bug dengan <code>setDnsSuffix</code> ketika kunci <code>SearchList</code> registri tidak ada.• Memperbaiki bug dalam logika devolusi DNS di. <code>setDnsSuffix</code>• Menambahkan rute jaringan setelah reboot perantara.• Mengizinkan <code>initializeVolume</code> untuk menulis ulang volume yang ada.• Menghapus informasi tambahan dari subperintah versi.	20 Juli 2021
2.0.285	<ul style="list-style-type: none">• Menambahkan opsi untuk menjalankan skrip pengguna dalam proses terpisah.• Userdata warisan (userdata XML) sekarang berjalan dalam proses terpisah, yang merupakan perilaku yang sama dengan agen peluncuran sebelumnya.• Menambahkan bendera CLI ke perintah <code>sysprep</code> dan <code>reset</code>, yang memungkinkannya untuk memblokir sampai layanan berhenti.• Membatasi izin folder konfigurasi.	8 Maret 2021

Versi	Detail	Tanggal rilis
2.0.207	<ul style="list-style-type: none">• Menambahkan bidang <code>hostName</code> opsional ke tugas <code>setHostName</code> .• Memperbaiki bug reboot. Reboot tugas <code>executeScript</code> dan <code>executeProgram</code> akan ditandai sebagai berjalan.• Menambahkan lebih banyak kode kembali untuk perintah <code>status</code>.• Menambahkan layanan <code>bootstrap</code> untuk memperbaiki masalah startup saat menjalankan tipe instans <code>t2.nano</code>.• Memperbaiki mode instalasi bersih untuk menghapus file yang tidak terlacak oleh installer.	2 Februari 2021
2.0.160	<ul style="list-style-type: none">• Memperbaiki perintah <code>validate</code> untuk mendeteksi nama tahap yang tidak valid.• Menambahkan perintah <code>w32tm resync</code> di tugas <code>addroutes</code> .• Memperbaiki masalah dengan mengubah urutan pencarian akhiran DNS.• Menambahkan syarat pemeriksaan agar bisa melaporkan data pengguna yang tidak valid dengan lebih baik.	4 Desember 2020
2.0.153	Menambahkan fungsionalitas <code>Sysprep</code> di <code>UserData</code>	3 November 2020

Versi	Detail	Tanggal rilis
2.0.146	<ul style="list-style-type: none"> • Memperbaiki masalah dengan RootExtend AMI non-Inggris. • Memberi izin menulis grup pengguna ke file log. • Membuat partisi MS Reserved untuk volume GPT. • Menambahkan perintah list-volumes dan tarik-turun volume di pengaturan Amazon EC2Launch. • Menambahkan get-agent-config perintah untuk mencetak file agent-config.yml dalam format yaml atau json. • Hapus kata sandi statis jika tidak ada kunci publik yang terdeteksi. 	6 Oktober 2020
2.0.124	<ul style="list-style-type: none"> • Menambahkan opsi untuk menampilkan versi OS pada wallpaper. • Menginisialisasi volume EBS yang dienkripsi. • Menambahkan rute untuk VPC tanpa nama DNS lokal. 	10 September 2020
2.0.104	<ul style="list-style-type: none"> • Membuat daftar pencarian sufiks DNS jika tidak ada. • Lewati mode Hibernasi jika tidak diminta. 	12 Agustus 2020
2.0.0	Pelepasan awal.	30 Juni 2020

Riwayat versi alat migrasi EC2Launch v2

Tabel berikut menjelaskan versi alat migrasi EC2Launch v2 yang dirilis.

Versi	Detail	Tanggal rilis
1.0.396	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru dari agen EC2launch v2:2.0.1924. 	Juni 11, 2024
1.0.394	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru agen EC2launch v2:2.0.1914. 	Juni 6, 2024
1.0.384	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru agen EC2launch v2:2.0.1881. 	8 Mei 2024
1.0.358	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru dari agen EC2launch v2:2.0.1815. 	8 Maret 2024
1.0.345	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru dari agen EC2launch v2:2.0.1739. 	Januari 18, 2024
1.0.342	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru dari agen EC2launch v2:2.0.1702. 	Januari 5, 2024
1.0.331	<ul style="list-style-type: none"> Perbarui alat migrasi dengan versi terbaru agen EC2Launch v2: 2.0.1643 Perbaiki kesalahan yang terjadi saat menjalankan <code>.Install.ps1 -DryRun</code>. Perbaiki masalah di mana konfigurasi kata sandi tidak disetel dengan benar <code>random</code> selama migrasi dari EC2config. Memperbaiki kesalahan yang terjadi jika <code>setWallpaper</code> disetel ke <code>False</code> selama migrasi dari EC2Launch. 	3 November 2023

Versi	Detail	Tanggal rilis
1.0.303	Perbarui alat migrasi dengan versi terbaru agen EC2Launch v2: 2.0.1580.	14 September 2023
1.0.286	Perbarui alat migrasi dengan versi terbaru agen EC2Launch v2: 2.0.1521.	14 Juli 2023
1.0.272	Perbarui alat migrasi dengan versi terbaru agen EC2Launch v2: 2.0.1303.	3 Mei 2023
1.0.262	Perbarui alat migrasi dengan versi terbaru agen EC2Launch v2: 2.0.1245.	9 Maret 2023
1.0.241	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.1011.	7 Desember 2022
1.0.218	<ul style="list-style-type: none">• Memvalidasi bahwa nilai Wilayah diambil dari metadata instans.• Memperbaiki bug kegagalan migrasi dalam paket bahasa.• Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.863.	3 September 2022
1.0.162	<ul style="list-style-type: none">• Memindahkan logika untuk menghapus agen lama ke EC2Launch v2 MSI.• Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.698.	18 Maret 2022
1.0.136	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.651.	13 Oktober 2021
1.0.130	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.548.	5 Agustus 2021
1.0.113	Menggunakan IMDSv2 sebagai pengganti IMDSv1.	4 Juni 2021

Versi	Detail	Tanggal rilis
1.0.101	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.285.	12 Maret 2021
1.0.86	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.207.	3 Februari 2021
1.0.76	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.160.	4 Desember 2020
1.0.69	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.153.	5 November 2020
1.0.65	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.146.	9 Oktober 2020
1.0.60	Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.124.	10 September 2020
1.0.54	<ul style="list-style-type: none">• Instal EC2Launch v2 jika tidak ada agen yang diinstal.• Menambahkan nomor versi file agen EC2Launch v2 ke 2.0.104.• Pisahkan agen SSM.	12 Agustus 2020
1.0.50	Menghapus NuGet ketergantungan.	10 Agustus 2020
1.0.0	Pelepasan awal.	30 Juni 2020

Konfigurasi instans Windows menggunakan EC2Launch

EC2Launch adalah seperangkat PowerShell skrip Windows yang menggantikan layanan EC2config pada AMI Windows Server 2016 dan 2019. Banyak dari AMI ini masih tersedia. EC2Launch v2 adalah agen peluncuran terbaru untuk semua versi Windows yang didukung, yang menggantikan

EC2Config dan EC2Launch. Untuk informasi selengkapnya, lihat [Konfigurasi instans Windows menggunakan EC2Launch v2](#).

 Note

Untuk menggunakan EC2Launch dengan IMDSv2, versinya harus 1.3.2002730 atau setelahnya.

Daftar Isi

- [Tugas EC2Launch](#)
- [Telemetri](#)
- [Instal EC2Launch versi terbaru](#)
- [Verifikasi versi EC2Launch](#)
- [Struktur direktori EC2Launch](#)
- [Konfigurasi EC2Launch](#)
- [Riwayat versi EC2Launch](#)

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 [Bekerja dengan data pengguna instans](#).
- Menetapkan rute statis persisten untuk mencapai layanan metadata dan AWS KMS server.

⚠ 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, lihat [Perbarui rute metadata/KMS untuk Server 2016 dan yang lebih baru saat meluncurkan AMI kustom](#).

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.

Untuk informasi selengkapnya tentang Windows Server 2019, lihat [Bandingkan Fitur di Versi Windows Server](#) di Microsoft.com.

Telemetri

Telemetri adalah informasi tambahan yang membantu AWS untuk lebih memahami kebutuhan Anda, 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
```

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 instans.
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 men-download 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 jendela PowerShell

```
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 `C:\ProgramData\Amazon\EC2-Windows\Launch`.

Verifikasi versi EC2Launch

Gunakan PowerShell perintah Windows berikut untuk memverifikasi versi EC2launch yang diinstal.

```
PS C:\> Test-ModuleManifest -Path "C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psd1" | Select Version
```

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.
- `Logs` — Berisi file log yang dihasilkan oleh skrip.

Versi **1.3.2004592** EC2launch dan yang lebih baru

Pengguna `Administrators` grup memiliki `Full control` izin untuk semua direktori EC2launch. Pengguna yang tidak berada dalam grup `Administrator` memiliki `Read & execute` izin untuk semua direktori EC2Launch kecuali `C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Config` direktori dibatasi untuk pengguna yang merupakan anggota `Administrators` grup.

EC2Launch versi **1.3.2004491** dan sebelumnya

Semua direktori EC2Launch mewarisi izin mereka dari, kecuali. C:\ProgramData C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Scripts Folder ini mewarisi semua izin awal dari C:\ProgramData saat dibuat, tetapi menghapus akses untuk pengguna normal ke CreateFiles dalam direktori.

Konfigurasi EC2Launch

Setelah instans Anda diinisialisasi untuk pertama kalinya, Anda dapat mengonfigurasi EC2Launch untuk berjalan lagi dan melakukan tugas start-up yang berbeda.

Tugas

- [Konfigurasi 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](#)

Konfigurasi 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 mengkonfigurasi sufiks standar lainnya. Untuk informasi selengkapnya tentang cara agen peluncuran mengatur sufiks DNS, lihat. [Konfigurasi DNS Suffix untuk agen peluncuran Windows](#)

```
region.ec2-utilities.amazonaws.com
```

- Perluas ukuran volume boot.
- Setel kata sandi administrator.

Untuk mengonfigurasi pengaturan inisialisasi

1. Saat ingin mengonfigurasi 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 EC2Launch 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.

```
PS C:\> 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:

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
SchedulePerBoot
```

2. Atau, jalankan executable dengan perintah berikut:

```
PS C:\> 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:

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

Untuk menginisialisasi disk setiap kali booting instans, tambahkan bendera -Schedule sebagai berikut:

```
PS C:\> 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. Konfigurasi 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 boot.

```
PS C:\> 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.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendWindowsIsReady.ps1 -
Schedule
```

Riwayat versi EC2Launch

AMI Windows, dimulai dengan Windows Server 2016, 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 riwayat [versi Windows AWS AMI](#).

Untuk mengunduh dan menginstal EC2Launch versi terbaru, lihat [Instal EC2Launch versi terbaru](#).

Tabel berikut menjelaskan versi EC2Launch yang dirilis. Perhatikan bahwa format versi berubah setelah versi 1.3.610.

Versi	Detail	Tanggal rilis
1.3.2004891	<ul style="list-style-type: none"> • Memperbaiki masalah yang tidak <code>HandleUserData</code> disetel <code>false</code> seperti yang diharapkan. • Menambahkan opsi <code>Encrypted</code> kata sandi ke <code>LaunchConfig.json</code> . • Mengubah <code>Settings</code> UI perilaku untuk mengenkripsi kata sandi yang ditentukan pengguna secara default. • Ditambahkan <code>SetAdminPasswordConfig.ps1</code> untuk mengonversi opsi <code>Specify</code> kata <code>Encrypted</code> sandi ke opsi kata sandi di file konfigurasi agen. 	31 Mei 2024
1.3.2004617	<ul style="list-style-type: none"> • Memperbaiki kesalahan saat mengatur wallpaper. 	15 Januari 2024
1.3.2004592	<ul style="list-style-type: none"> • Izin akses yang diperbarui yang ditetapkan oleh <code>install.ps1</code> untuk <code>%ProgramData%\Amazon\EC2-Windows\Launch</code> . • Membatasi akses folder/file EC2Launch untuk baca-eksekusi saja pada akun pengguna standar. • 	2 Januari 2024

Versi	Detail	Tanggal rilis
	<p>Mengubah agen agar berhenti menunggu Layanan Metadata Instans (IMDS) untuk diinisialisasi jika IMDS tidak diaktifkan untuk instans.</p> <ul style="list-style-type: none"> • Menambahkan batas waktu lima menit saat menunggu IMDS diinisialisasi. • Mengubah agen untuk menulis telemetri ke log konsol instans sebelum pesan <code>Windows is Ready</code>, bukan setelahnya. • Menambahkan dukungan wallpaper ke beberapa tipe instans baru. <p>Untuk informasi selengkapnya tentang izin akses dan izin akun pengguna dari direktori EC2Launch, lihat. the section called “Struktur direktori EC2Launch”</p>	
1.3.2004491	<ul style="list-style-type: none"> • Menambahkan telemetri untuk memantau penggunaan opsi Tentukan kata sandi admin. 	9 November 2023
1.3.2004462	<ul style="list-style-type: none"> • Menambahkan flush setelah setiap penulisan ke konsol serial. 	18 Oktober 2023
1.3.2004438	<ul style="list-style-type: none"> • Membatasi devolusi nama domain berdasarkan entri registri: <code>HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel</code> . • Izin <code>UserdataExecution.log</code> terbatas hanya untuk <code>Administrators</code> . • Menambahkan pesan kesalahan di Log Peristiwa Windows saat inialisasi log gagal. 	4 Oktober 2023

Versi	Detail	Tanggal rilis
1.3.2004256	<ul style="list-style-type: none"> Menambahkan nilai <code>EnableSCSIPersistentReservations</code> ke log konsol. Menambahkan kemampuan coba lagi untuk <code>Get-ConsolePort</code>. 	7 Juli 2023
1.3.2004052	<ul style="list-style-type: none"> Memperbaiki kesalahan yang terjadi saat tidak ada kunci SSH yang ditentukan saat peluncuran instans. Diperbarui untuk mencoba lagi memulai layanan AmazonSSM Agent Windows saat mengalami kegagalan. Diperbarui untuk <code>SysprepInstance</code> gagal.ps1 jika <code>BeforeSysprep .cmd</code> gagal dengan kode keluar bukan nol. 	8 Maret 2023
1.3.2003975	<ul style="list-style-type: none"> Memperbaiki masalah yang memengaruhi build Packer AMI di mana <code>SysprepInstance.ps1</code> mengembalikan <code>1. \$LastErrorCode</code> 	24 Desember 2022
1.3.2003961	<ul style="list-style-type: none"> Memperbaiki masalah di mana kata sandi administrator yang ditentukan secara eksplisit ditimpa dengan kata sandi acak pada instans yang diluncurkan dengan cepat. Memperbaiki masalah di mana Agen SSM gagal memulai pada tipe instans yang lebih kecil. Memperbaiki masalah saat log konsol instans berisi, <code>RPCERTIFICATE-THUMBPRINT: 00000000000000000000000000000000</code> bukan nilai cap jempol sertifikat RDP yang valid. 	6 Desember 2022
1.3.2003923	<ul style="list-style-type: none"> Memperbaiki logika untuk menemukan adaptor jaringan ketika <code>PnPDeviceID</code> kosong. 	9 November 2022

Versi	Detail	Tanggal rilis
1.3.2003919	<ul style="list-style-type: none">• Diperbarui Dapatkan- ConsolePort untuk menggunakan informasi segmen PCI.• Memperbaiki masalah di mana adaptor jaringan yang salah dapat dipilih setelah reboot.• Logika batas waktu start-SSM-agent tetap.• Memperbaiki kompatibilitas mundur untuk alias AdminCredentials fungsi Kirim.	8 November 2022
1.3.2003857	<ul style="list-style-type: none">• Memprioritaskan adaptor dengan gateway default saat adaptor jaringan utama dipilih.• Enkripsi kata sandi dalam memori yang diperluas.	3 Oktober 2022
1.3.2003824	<ul style="list-style-type: none">• Memperbaiki kesalahan selama setComputerName .• Menambahkan logika untuk melewati aktivasi Windows ketika kode penagihan BYOL terdeteksi.• Menambahkan enkripsi kata sandi dalam memori.• Memperbaiki kesalahan selama inisialisasi volume aktif. m6id.4xlarge	30 Agustus 2022
1.3.2003691	<ul style="list-style-type: none">• Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja.• Memperbaiki bug yang memengaruhi instalasi eGPU.	21 Juni 2022
1.3.2003639	<ul style="list-style-type: none">• Menambahkan logika tunggu adaptor jaringan untuk mencegah penggunaan sebelum inisialisasi.• Memperbaiki masalah kecil.	10 Mei 2022

Versi	Detail	Tanggal rilis
1.3.2003498	<ul style="list-style-type: none"> Menambahkan telemetri. Menambahkan pintasan ke UI Pengaturan. PowerShell Skrip yang diformat. 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 Instal- EgpuManager dengan dukungan IMDSv2.	7 Juni 2021
1.3.2003312	<ul style="list-style-type: none"> Menambahkan baris log sebelum dan sesudah pengaturan <code>setMonitorAlwaysOn</code> . Menambahkan versi paket AWS Nitro Enclave ke log konsol. 	4 Mei 2021
1.3.2003284	Peningkatan model izin dengan memperbarui lokasi untuk menyimpan data pengguna ke <code>LocalAppData</code> .	23 Maret 2021
1.3.2003236	<ul style="list-style-type: none"> Metode yang diperbarui untuk mengatur kata sandi pengguna di <code>Set-AdminAccount</code> dan <code>Randomize-LocalAdminPassword</code> . Memperbaiki <code>InitializeDisks</code> untuk memeriksa apakah disk diatur untuk membaca hanya sebelum mengaturnya menjadi dapat ditulis. 	11 Februari 2021
1.3.2003210	Perbaiki lokalisasi untuk <code>install.ps1</code> .	7 Januari 2021
1.3.2003205	Perbaiki keamanan untuk <code>install.ps1</code> untuk memperbarui izin di direktori <code>%ProgramData%AmazonEC2-WindowsLaunchModuleScripts</code> .	28 Desember 2020
1.3.2003189	Menambahkan <code>w32tm resync</code> setelah menambahkan rute.	4 Desember 2020

Versi	Detail	Tanggal rilis
1.3.2003155	Informasi tipe instans yang diperbarui.	25 Agustus 2020
1.3.2003150	Menambahkan <code>OsCurrentBuild</code> dan <code>OsReleaseId</code> 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
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 <code>InitializeDisks</code> dengan.ps1 di mana menjalankan skrip pada node di Windows Server Failover Cluster 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	<ul style="list-style-type: none"> Menambahkan plugin untuk mengatur monitor agar tidak pernah mati untuk memperbaiki masalah ACPI. Edisi dan versi SQL Server ditulis ke konsol. 	21 Januari 2019
1.3.2000930	Perbaikan untuk menambahkan rute ke metadata pada ENI <code>ipv6-enabled</code> .	2 Januari 2019

Versi	Detail	Tanggal rilis
1.3.2000760	<ul style="list-style-type: none"> Menambahkan konfigurasi default untuk pengaturan RSS dan Menerima Antrean untuk perangkat ENA. Hibernasi dinonaktifkan selama Sysprep. 	5 Desember 2018
1.3.2000630	<ul style="list-style-type: none"> Menambahkan rute 169.254.169.253/32 untuk server DNS. Menambahkan filter pengaturan pengguna Admin. Perbaikan dilakukan pada hibernasi instans. Menambahkan opsi untuk menjadwalkan EC2Launch yang akan dijalankan di setiap boot. 	9 November 2018
1.3.2000430.0	<ul style="list-style-type: none"> Menambahkan rute 169.254.169.123/32 ke layanan waktu AMZN. Menambahkan rute 169.254.169.249/32 ke layanan lisensi GRID. Menambahkan batas waktu habis 25 detik saat mencoba memulai Systems Manager. 	19 September 2018
1.3.200039.0	<ul style="list-style-type: none"> Memperbaiki huruf drive yang tidak tepat untuk volume EBS NVME. 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	<ul style="list-style-type: none"> Menambahkan tipe instans yang hilang di wallpaper. Memperbaiki masalah pemetaan huruf drive dan penginstalan disk. 	

Versi	Detail	Tanggal rilis
1.3.580	<ul style="list-style-type: none">• Memperbaiki Get-Metadata untuk menggunakan pengaturan proxy sistem default untuk permintaan web.• Kasus khusus untuk NVMe dalam inisialisasi disk ditambahkan.• 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	<ul style="list-style-type: none">• Memperbaiki masalah panjang heksadesimal untuk perubahan nama komputer.• Memperbaiki kemungkinan loop reboot untuk perubahan nama komputer.• Memperbaiki masalah dalam pengaturan wallpaper.	
1.2.0	<ul style="list-style-type: none">• Perbarui untuk menampilkan informasi tentang sistem operasi (OS) yang diinstal di log sistem EC2.• Perbarui untuk menampilkan versi EC2Launch dan SSM Agent di log sistem EC2.• Memperbaiki masalah kecil.	

Versi	Detail	Tanggal rilis
1.1.2	<ul style="list-style-type: none">• Perbarui untuk menampilkan informasi driver ENA di log sistem EC2.• Perbarui untuk mengecualikan Hyper-V dari logika filter NIC utama.• Menambahkan AWS KMS server dan port ke kunci registri untuk aktivasi KMS.• Penyiapan wallpaper yang ditingkatkan untuk banyak pengguna.• Perbarui untuk menghapus rute dari penyimpanan persisten.• Perbarui untuk menghapus z dari zona ketersediaan di daftar sufiks DNS.• Perbarui untuk mengatasi masalah dengan tag <run AsLocal System> di data pengguna.	
1.1.1	Pelepasan awal.	

Konfigurasi instance Windows menggunakan layanan EC2config (legacy)

Note

Dokumentasi EC2config disediakan hanya untuk referensi historis. Versi sistem operasi yang dijelankannya tidak lagi didukung oleh Microsoft. Kami sangat menyarankan Anda meningkatkan ke layanan peluncuran terbaru.

Layanan peluncuran terbaru untuk Windows Server 2022 adalah [EC2Launch v2](#), yang menggantikan EC2Config dan EC2Launch.

Windows AMI 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.

Note

EC2Launch menggantikan EC2Config di AMI Windows untuk Windows Server 2016 dan 2019. Untuk informasi selengkapnya, lihat [Konfigurasi instans Windows menggunakan EC2Launch](#). Layanan peluncuran terbaru untuk semua versi Windows Server yang didukung adalah [EC2Launch v2](#), yang menggantikan EC2Config dan EC2Launch.

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

- [Tugas EC2Config](#)
- [Menginstal EC2Config versi terbaru](#)
- [Hentikan, mulai ulang, hapus, atau uninstal EC2Config](#)
- [EC2config dan AWS Systems Manager](#)
- [EC2Config dan Sysprep](#)
- [Properti layanan EC2](#)
- [File pengaturan EC2Config](#)
- [Konfigurasi pengaturan proxy untuk layanan EC2Config](#)
- [Riwayat versi EC2Config](#)
- [Pemecahan masalah layanan EC2Config](#)

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 sudah diinstal). Untuk informasi selengkapnya tentang menentukan data pengguna, lihat [Bekerja dengan data pengguna instans](#).

EC2Config melakukan tugas-tugas berikut setiap kali instans dimulai:

- Ubah nama host agar sesuai dengan alamat IP privat dalam notasi Hex (tugas ini dinonaktifkan secara default dan harus diaktifkan untuk dijalankan saat dimulainya instans).
- Konfigurasi 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 agar dapat dijalankan saat instans dimulai).
- 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::240] 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 dengan Windows Sysprep](#).

Menginstal EC2Config versi terbaru

Secara default, file EC2Config sudah disertakan dalam AMI sebelum Windows Server 2016. Ketika layanan EC2config diperbarui, AMI Windows baru dari AWS menyertakan versi terbaru layanan. Namun, Anda perlu memperbarui AMI Windows Anda sendiri dan instans dengan EC2Config versi terbaru.

Note

EC2Launch menggantikan EC2Config di Windows Server 2016 dan 2019. Untuk informasi selengkapnya, lihat [Konfigurasi instans Windows menggunakan EC2Launch](#). Layanan peluncuran terbaru untuk semua versi Windows Server yang didukung adalah [EC2Launch v2](#), yang menggantikan EC2Config dan EC2Launch.

Untuk informasi tentang cara menerima notifikasi pembaruan EC2Config, lihat [Berlangganan notifikasi layanan EC2Config](#). 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.
- Jika versi EC2Config Anda lebih awal dari versi 2.1.19 dan Anda menginstal versi 2.2.12 atau sebelumnya, maka Anda harus menginstal versi 2.1.19 terlebih dahulu. Untuk menginstal versi 2.1.19, unduh [EC2Install_2.1.19.zip](#), buka file zip, lalu jalankan `EC2Install.exe`.

 Note

Jika versi EC2Config Anda lebih awal dari versi 2.1.19 dan Anda menginstal versi 2.3.313 atau setelahnya, maka Anda dapat menginstalnya secara langsung tanpa perlu menginstal versi 2.1.19 terlebih dahulu.

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 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.

Hentikan, mulai ulang, hapus, atau uninstal EC2Config

Anda dapat mengelola layanan EC2Config sama seperti yang Anda lakukan pada layanan lainnya.

Untuk menerapkan pengaturan yang diperbarui ke instans, Anda dapat menghentikan dan memulai ulang layanan. Jika Anda menginstal EC2Config secara manual, maka Anda harus menghentikan layanan terlebih dahulu.

Untuk menghentikan layanan EC2Config

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, arahkan ke Alat Administratif, lalu klik Layanan.
3. Di daftar layanan, klik kanan EC2Config, lalu pilih Berhenti.

Untuk memulai ulang layanan EC2Config

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, arahkan ke Alat Administratif, lalu klik Layanan.
3. Di daftar layanan, klik kanan EC2Config, lalu pilih Mulai ulang.

Jika Anda tidak perlu memperbarui pengaturan konfigurasi, membuat AMI Anda sendiri, atau menggunakan, AWS Systems Manager Anda dapat menghapus dan mencopot pemasangan layanan. Menghapus layanan akan menghapus subkunci registri. Menghapus instalasi layanan akan menghapus file, subkunci registri, dan pintasan apa pun ke layanan tersebut.

Untuk menghapus layanan EC2Config

1. Mulai jendela prompt perintah.
2. Jalankan perintah berikut:

```
sc delete ec2config
```

Untuk melepas instalasi EC2Config

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pada menu Mulai, klik Panel Kontrol.
3. Klik dua kali Program dan Fitur.

4. Pada daftar program, pilih EC2 ConfigService, dan klik Uninstall.

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 [Perbarui SSM Agent dengan menggunakan Run Command](#) dalam Panduan Pengguna AWS Systems Manager .

EC2Config dan Sysprep

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 ini secara tidak langsung menggunakan kotak dialog Properti Layanan EC2, atau langsung menggunakan editor XML atau editor teks. Namun, ada beberapa pengaturan lanjutan yang tidak tersedia di kotak dialog 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 AMI yang didukung Amazon EBS-Backed](#).

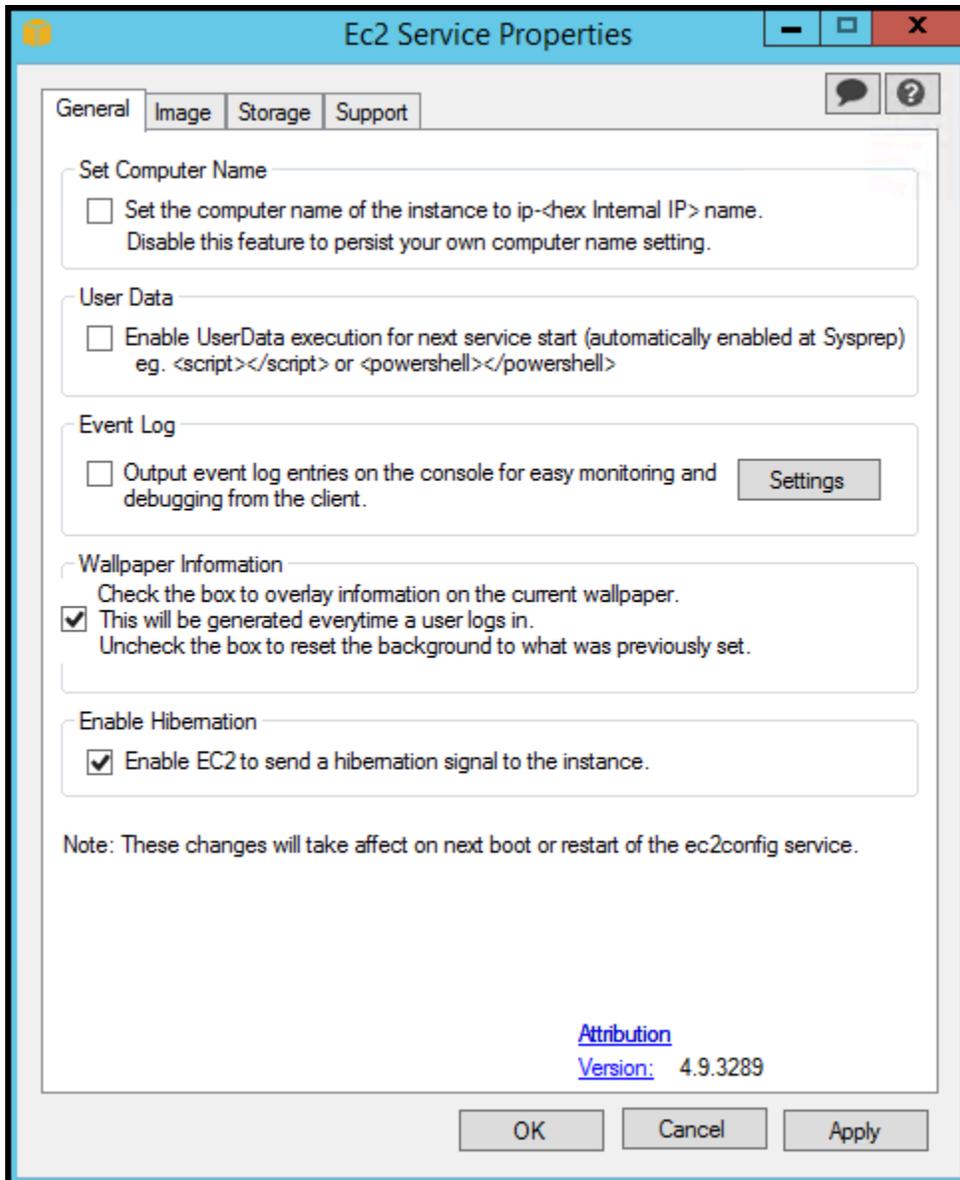
Properti layanan EC2

Prosedur berikut menjelaskan cara menggunakan kotak dialog Properti Layanan Ec2 untuk mengaktifkan atau menonaktifkan pengaturan.

Untuk mengubah pengaturan menggunakan kotak dialog Properti Layanan Ec2

1. Jalankan dan hubungkan ke instans Windows Anda.

2. Dari menu Start, klik Semua Program, dan kemudian klik ConfigServicePengaturan EC2.



3. Pada tab Umum dari kotak dialog 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.

Klik 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.

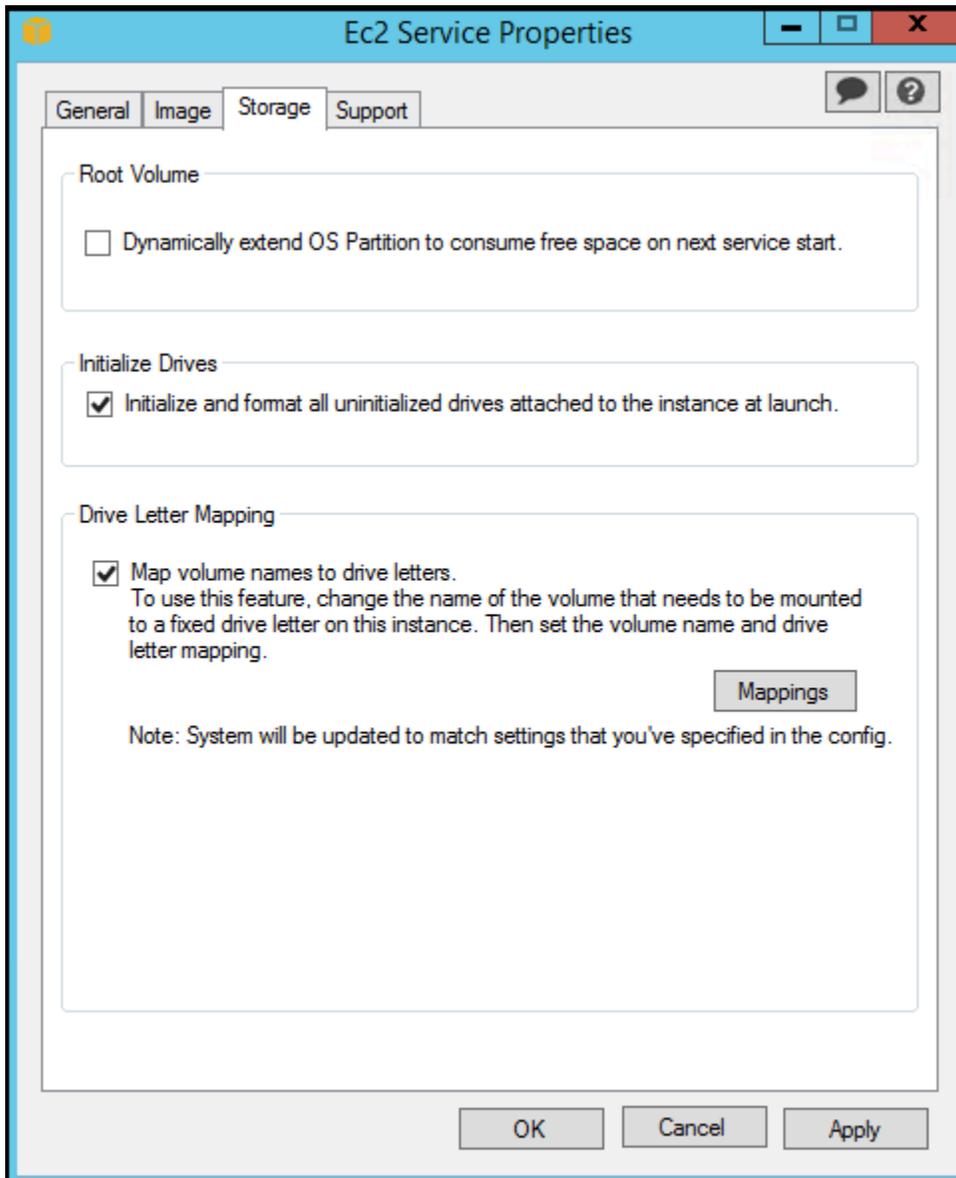
```
Hostname      : WIN-U0RFOJCTPUU
Instance ID   : i-d583f76a
Public IP Address : 54.208.43.227
Private IP Address : 172.31.42.195
Availability Zone : us-east-1b
Instance Size  : t2.micro
Architecture   : AMD64
```

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. Klik 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. Pengaturan ini dapat berguna ketika instans di-boot dari volume perangkat root yang memiliki ukuran khusus.

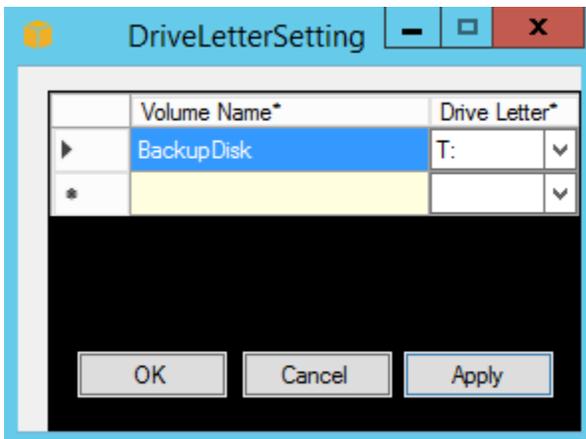
Inisialisasi Drive

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, klik Pemetaan. Di kotak dialog DriveLetterPengaturan, tentukan Nilai Nama Volume dan Huruf Drive untuk setiap volume, klik Terapkan, lalu klik 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 melanjutkan pengerjaannya nanti, klik OK untuk menutup kotak dialog Properti Layanan EC2. Jika Anda telah selesai menyesuaikan instans Anda dan ingin membuat AMI dari instans itu, lihat [Buat AMI dengan 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 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 file. AWS KMS AWS KMS harus berada di Wilayah yang sama dengan instans Anda.
- `DiscoverFromZone`—Menemukan AWS KMS server dari zona DNS yang ditentukan.
- `ReadFromUserData` Mendapatkan 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 contoh 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-in

- `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 di file ini, lihat [Kunci Eventlog](#) di Pustaka MSDN.
- `Ec2ConfigureRDP`—Menyiapkan sertifikat yang ditandatangani sendiri di instans, sehingga pengguna dapat mengakses instans dengan aman menggunakan Remote Desktop. Pengaturan ini tidak digunakan untuk contoh 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 dalam tag skrip disimpan ke file batch, dan perintah yang dibungkus PowerShell tag disimpan ke file.ps1 (sesuai dengan kotak centang Data Pengguna pada kotak dialog 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. Hanya didukung untuk fitur Microsoft Hyper-V dan layanan vmms yang sesuai.

Pengaturan Global

- `ManageShutdown`—Memastikan bahwa instans yang diluncurkan dari AMI yang didukung penyimpanan instans tidak berhenti 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 mengkonfigurasi sufiks standar lainnya. Untuk informasi selengkapnya tentang cara agen peluncuran mengatur sufiks DNS, lihat [Konfigurasi DNS Suffix untuk agen peluncuran 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.
- **RemoveCredentialsfromSysprepStartup**—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 contoh, *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 contoh, *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:

FormatWithMEMANGKAS

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.

FormatWithoutMEMANGKAS

Menonaktifkan perintah TRIM saat memformat drive dan meningkatkan waktu pemformatan di Windows. Setelah drive diformat dan diinisialisasi, sistem memulihkan konfigurasi TRIM.

DisableInitializeDrive

Menonaktifkan pemformatan untuk drive baru. Gunakan pengaturan ini untuk memulai drive secara manual.

Konfigurasi pengaturan proxy untuk layanan EC2Config

Anda dapat mengonfigurasi layanan EC2config untuk berkomunikasi melalui proxy menggunakan salah satu metode berikut: SDK for AWS .NET, system.net elemen, atau Kebijakan Grup Microsoft dan Internet Explorer. Menggunakan AWS SDK for .NET adalah metode yang lebih disukai karena Anda dapat menentukan kredensi login.

Metode

- [Konfigurasi pengaturan proxy menggunakan AWS SDK for .NET \(Preferred\)](#)
- [Konfigurasi pengaturan proxy menggunakan elemen system.net](#)
- [Konfigurasi pengaturan proxy menggunakan Kebijakan Grup Microsoft dan Internet Explorer Microsoft](#)

Konfigurasi pengaturan proxy menggunakan AWS SDK for .NET (Preferred)

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 for .NET](#).

Untuk menentukan elemen proxy di `Ec2Config.exe.config`

1. Edit file `Ec2Config.exe.config` pada instans di mana Anda menginginkan file layanan EC2Config untuk berkomunikasi melalui proxy. 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.

Konfigurasi 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 Proxy default \(Pengaturan Jaringan\)](#) di MSDN.

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 Proxy default \(Pengaturan Jaringan\)](#) di MSDN.

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" />
  </bypasslist>
</defaultProxy>
```

```
<add address="169.254.169.254" />
<add address="[fd00:ec2::250]" />
<add address="[fd00:ec2::254]" />
</bypasslist>
</defaultProxy>
```

3. Simpan perubahan Anda.

Konfigurasi 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.

Riwayat versi EC2Config

AMI Windows sebelum Windows Server 2016 menyertakan layanan opsional yang disebut layanan EC2Config (`EC2Config.exe`). EC2Config dimulai saat instans melakukan boot dan menjalankan tugas selama pemulaian dan setiap kali Anda menghentikan atau memulai instans.

Anda dapat menerima notifikasi saat layanan EC2Config versi baru dirilis. Untuk informasi selengkapnya, lihat [Berlangganan notifikasi layanan EC2Config](#).

Tabel berikut menjelaskan versi EC2Config yang dirilis. Untuk informasi tentang pembaruan untuk SSM Agent, lihat [Catatan Rilis Systems Manager SSM Agent](#).

Versi	Detail	Tanggal rilis
4.9.5554	<ul style="list-style-type: none"> Membatasi devolusi nama domain berdasarkan entri registri: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel . Versi baru SSM Agent 3.2.1630.0 . 	4 Oktober 2023
4.9.5467	<ul style="list-style-type: none"> Menambahkan kemampuan coba lagi untuk menemukan port konsol. Versi baru SSM Agent 3.1.2282.0 . 	1 Agustus 2023
4.9.5288	<ul style="list-style-type: none"> AWS Core SDK yang diperbarui ke versi 3.7.103.23 . Memperbaiki masalah saat dokumen AWS-UpdateEC2Config SSM gagal diperbarui EC2Config pada instans yang diaktifkan hanya dengan IMDSv2. Versi baru SSM Agent 3.1.2144.0 . 	8 Maret 2023
4.9.5231	<ul style="list-style-type: none"> Versi baru SSM Agent 3.1.1927.0. 	14 Februari 2023
4.9.5103	<ul style="list-style-type: none"> Memperbaiki masalah di mana volume fana diidentifikasi secara tidak benar pada keluarga instans r5d dan i4i. Versi baru SSM Agent 3.1.1856.0. 	5 Desember 2022

Versi	Detail	Tanggal rilis
4.9.5064	<ul style="list-style-type: none"> • Diperbarui untuk menggunakan informasi segmen PCI guna memilih port konsol. • PowerShell Skrip yang ditandatangani dan menambahkan header hak cipta. • Logika pemilihan adaptor jaringan utama tetap. • Versi baru SSM Agent 3.1.1732.0. 	16 November 2022
4.9.4588	<ul style="list-style-type: none"> • Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja. • Menambahkan pustaka bersama agen peluncuran libec2launch.dll. • Versi baru SSM Agent 3.1.1188.0. 	31 Mei 2022
4.9.4556	<ul style="list-style-type: none"> • Menambahkan logika tunggu untuk memastikan inisialisasi penuh NIC sebelum digunakan. • Versi baru Log4Net 2.0.14.0 mengambil patch keamanan. • Versi baru SSM Agent 3.1.1045.0 mengambil patch keamanan. 	1 Maret 2022
4.9.4536	<ul style="list-style-type: none"> • Memperbaiki masalah saat data pengguna mogok saat folder Temp hilang. • Versi baru SSM Agent 3.1.804.0. 	31 Januari 2022
4.9.4508	<ul style="list-style-type: none"> • Memperbaiki masalah untuk menghitung jalur skrip diskpart dengan benar. • Versi baru SSM Agent 3.1.338.0. 	6 Oktober 2021

Versi	Detail	Tanggal rilis
4.9.4500	<ul style="list-style-type: none"> • <code>Install-EgpuManagerConfig</code> yang diperbarui dengan dukungan IMDS v2. • Tautan web yang diperbarui untuk menggunakan https. • Versi baru SSM Agent 3.1.282.0 	7 September 2021
4.9.4419	<ul style="list-style-type: none"> • Memperbaiki logika fallback IMDS versi 1 • Memperbarui semua penggunaan direktori temp Windows untuk direktori temp EC2Config • Versi baru SSM Agent 3.0.1124.0 	2 Juni 2021
4.9.4381	<ul style="list-style-type: none"> • Ditambahkan dukungan untuk skema dokumen SSM versi 2.2 di EC2 ConfigUpdater • Menambahkan versi paket AWS Nitro Enclave ke log konsol • Versi baru SSM Agent 3.0.529.0 	4 Mei 2021
4.9.4326	<ul style="list-style-type: none"> • Menghapus semua tautan di pengaturan UI • Ini adalah versi EC2Config terakhir yang mendukung Windows Server 2008. 	3 Maret 2021
4.9.4279	<ul style="list-style-type: none"> • Memperbaiki masalah keamanan yang terkait dengan tugas terjadwal <code>Ec2ConfigMonitor</code> • Memperbaiki masalah pemetaan huruf drive dan jumlah disk ephemeral yang salah • Menambahkan <code>OsCurrentBuild</code> dan <code>OsReleaseId</code> ke output konsol • Versi baru SSM Agent 2.3.871.0 	11 Desember 2020
4.9.4222	<ul style="list-style-type: none"> • Memperbaiki logika fallback IMDS versi 1 • Versi baru SSM Agent 2.3.842.0 	7 April 2020

Versi	Detail	Tanggal rilis
4.9.4122	<ul style="list-style-type: none"> Menambahkan dukungan untuk IMDS V2 Versi baru SSM Agent 2.3.814.0 	4 Maret 2020
4.9.3865	<ul style="list-style-type: none"> Memperbaiki masalah pendeteksian port COM untuk Windows Server 2008 R2 pada instans metal Versi baru SSM Agent 2.3.722.0 	31 Oktober 2019
4.9.3519	<ul style="list-style-type: none"> Versi baru SSM Agent 2.3.634.0 	18 Juni 2019
4.9.3429	<ul style="list-style-type: none"> Versi baru SSM Agent 2.3.542.0 	25 April 2019
4.9.3289	<ul style="list-style-type: none"> Versi baru 2.3.444.0 	11 Februari 2019
4.9.3270	<ul style="list-style-type: none"> Menambahkan plugin untuk mengatur monitor agar tidak pernah mati untuk memperbaiki masalah ACPI Edisi dan versi SQL Server yang ditulis ke konsol Versi baru SSM Agent 2.3.415.0 	22 Januari 2019
4.9.3230	<ul style="list-style-type: none"> Deskripsi Pemetaan Huruf Drive yang diperbarui agar lebih selaras dengan fungsionalitas Versi baru SSM Agent 2.3.372.0 	10 Januari 2019
4.9.3160	<ul style="list-style-type: none"> Peningkatan waktu tunggu untuk NIC utama Menambahkan konfigurasi default untuk pengaturan RSS dan Menerima Antrean untuk perangkat ENA Hibernasi dinonaktifkan selama Sysprep Versi baru SSM Agent 2.3.344.0 AWS SDK yang ditingkatkan ke 3.3.29.13 	15 Desember 2018
4.9.3067	<ul style="list-style-type: none"> Perbaikan yang dilakukan pada hibernasi instans Versi baru SSM Agent 2.3.235.0 	8 November 2018
4.9.3034	<ul style="list-style-type: none"> Menambahkan rute 169.254.169.253/32 untuk server DNS Versi baru SSM Agent 2.3.193.0 	24 Oktober 2018

Versi	Detail	Tanggal rilis
4.9.2986	<ul style="list-style-type: none"> Menambahkan penandatanganan untuk semua biner terkait EC2Config 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	<ul style="list-style-type: none"> Versi baru SSM Agent (2.3.50.0) Menambahkan rute 169.254.169.123/32 ke layanan waktu AMZN Menambahkan rute 169.254.169.249/32 ke layanan lisensi GRID 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
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

Versi	Detail	Tanggal rilis
4.9.2565	<ul style="list-style-type: none">Versi baru SSM Agent (2.2.355.0)Memperbaiki masalah pada instans M5 dan C5 (tidak dapat menemukan driver PV)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
4.9.2327	<ul style="list-style-type: none">Versi baru SSM Agent (2.2.120.0)Penambahan penemuan port COM pada instans bare metal Amazon EC2Penambahan logging status Hyper-V 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

Versi	Detail	Tanggal rilis
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	<ul style="list-style-type: none"> Versi baru SSM Agent (2.2.24.0) 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)	
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	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.879.0) Memperbaiki jalur direktori CloudWatch Log untuk Windows Server 2003 	

Versi	Detail	Tanggal rilis
4.9.1981	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.847.0) Memperbaiki masalah dengan pembuatan <code>important.txt</code> sedang di volume EBS. 	
4.9.1964	Versi baru SSM Agent (2.0.842.0)	
4.9.1951	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.834.0) Memperbaiki masalah huruf drive yang tidak dipetakan dari Z: untuk drive singkat. 	
4.9.1925	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.822.0) [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)	
4.9.1876	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.796.0) Memperbaiki masalah keluaran/pengalihan error untuk eksekusi data pengguna admin. 	
4.9.1863	<ul style="list-style-type: none"> Versi baru SSM Agent (2.0.790.0) Memperbaiki masalah terkait pemasangan beberapa volume EBS ke instans Amazon EC2. Ditingkatkan CloudWatch untuk mengambil jalur konfigurasi, menjaga kompatibilitas mundur. 	
4.9.1791	Versi baru SSM Agent (2.0.767.0)	

Versi	Detail	Tanggal rilis
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	<ul style="list-style-type: none">Versi baru SSM Agent (2.0.672.0)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)	
4.2.1442	Versi baru SSM Agent (2.0.599.0)	
4.1.1378	Versi baru SSM Agent (2.0.558.0)	

Versi	Detail	Tanggal rilis
4.0.1343	<ul style="list-style-type: none">• Run Command, State Manager, CloudWatch agen, dan dukungan domain join telah dipindahkan ke agen lain yang disebut SSM Agent. SSM Agent akan diinstal sebagai bagian dari pemutakhiran EC2Config. Untuk informasi selengkapnya, lihat EC2config dan AWS Systems Manager.• 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: Menginstall dan Mengonfigurasi SSM Agent pada Instans Windows di Panduan Pengguna AWS Systems Manager .• Jika sebelumnya Anda mengaktifkan CloudWatch integrasi pada instance Anda dengan menggunakan file konfigurasi lokal (<code>AWS.EC2.Windows.CloudWatch.json</code>), Anda harus mengonfigurasi file agar berfungsi dengan Agen SSM.	
3.19.1153	<ul style="list-style-type: none">• Plugin aktivasi yang diaktifkan kembali untuk instance dengan konfigurasi lama AWS KMS . Lewati aktivasi untuk pengguna BYOL.• Ubah perilaku TRIM default untuk dinonaktifkan selama operasi format disk dan tambahkan FormatWith TRIM untuk mengganti InitializeDisks plugin dengan data pengguna.	

Versi	Detail	Tanggal rilis
3.18.1118	<ul style="list-style-type: none"> Perbaiki untuk menambahkan rute dengan andal ke adaptor jaringan utama. Pembaruan untuk meningkatkan dukungan untuk AWS layanan. 	
3.17.1032	<ul style="list-style-type: none"> Memperbaiki log sistem duplikat yang muncul saat filter disetel ke kategori yang sama. 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	<p>Menambahkan dukungan untuk mengganti pengaturan InitializeDisks plugin. Contoh: Untuk mempercepat inisialisasi disk SSD, Anda dapat menonaktifkan TRIM untuk sementara dengan menentukan ini di userdata:</p> <pre>< InitializeDrives Pengaturan>< > TRIM </ ></ SettingsGroup Pengaturan FormatWithout SettingsGroup InitializeDrives</pre>	
3.13.727	System Manager Run Command - Perbaikan untuk memproses perintah dengan andal setelah windows reboot.	

Versi	Detail	Tanggal rilis
3.12.649	<ul style="list-style-type: none">• Perbaikan untuk menangani booting ulang dengan baik saat menjalankan perintah/skrip.• Perbaiki untuk membatalkan perintah yang berjalan dengan andal.• Tambahkan dukungan untuk (secara opsional) mengunggah log MSI ke S3 saat menginstal aplikasi melalui Systems Manager Run Command.	
3.11.521	<ul style="list-style-type: none">• Perbaikan untuk mengaktifkan pembuatan sidik jari RDP untuk Windows Server 2003.• Perbaikan untuk menyertakan perbedaan zona waktu dan UTC di baris log EC2Config.• Dukungan Systems Manager untuk menjalankan perintah Run Command secara paralel.• Kembalikan perubahan sebelumnya untuk menghadirkan disk yang dipartisi secara online.	
3.10.442	<ul style="list-style-type: none">• Memperbaiki kegagalan konfigurasi Systems Manager saat menginstal aplikasi MSI.• Perbaiki untuk menghadirkan disk penyimpanan online dengan andal.• Pembaruan untuk meningkatkan dukungan untuk AWS layanan.	

Versi	Detail	Tanggal rilis
3.9.359	<ul style="list-style-type: none">• Perbaiki dalam skrip pasca Sysprep untuk membiarkan konfigurasi pembaruan windows dalam status default.• Perbaiki plugin pembuat kata sandi untuk meningkatkan keandalan dalam mendapatkan pengaturan kebijakan kata sandi GPO.• Membatasi/Izin folder log EC2Config/SSM ke grup Administrator lokal.• Pembaruan untuk meningkatkan dukungan untuk AWS layanan.	
3.8.294	<ul style="list-style-type: none">• Memperbaiki masalah CloudWatch yang mencegah log diunggah saat tidak di drive utama.• Meningkatkan proses inisialisasi disk dengan menambahkan logika coba lagi.• Menambahkan penanganan kesalahan yang ditingkatkan saat SetPassword plugin terkadang gagal selama pembuatan AMI.• Pembaruan untuk meningkatkan dukungan untuk AWS layanan.	

Versi	Detail	Tanggal rilis
3.7.308	<ul style="list-style-type: none">• Perbaikan pada utilitas ec2config-cli untuk pengujian konfigurasi dan pemecahan masalah dalam instans.• Hindari menambahkan rute statis untuk AWS KMS dan layanan meta-data pada adaptor OpenVPN.• Memperbaiki masalah ketika eksekusi data pengguna tidak mematuhi tag "persisten".• Penanganan kesalahan yang lebih baik saat masuk ke konsol EC2 tidak tersedia.• Pembaruan untuk meningkatkan dukungan untuk AWS layanan.	
3.6.269	<ul style="list-style-type: none">• Perbaikan keandalan aktivasi Windows untuk pertama kali menggunakan alamat lokal tautan 169.254.0.250/251 untuk mengaktifkan windows melalui AWS KMS• Penanganan proxy yang lebih baik untuk skenario Systems Manager, Aktivasi Windows, dan Penggabungan Domain• Memperbaiki masalah di mana baris duplikat akun pengguna ditambahkan ke file jawaban Sysprep	
3.5.228	<ul style="list-style-type: none">• Mengatasi skenario di mana CloudWatch plugin dapat mengkonsumsi CPU yang berlebihan dan membaca memori Windows Event Logs• Menambahkan tautan ke dokumentasi CloudWatch konfigurasi di UI Pengaturan EC2config	

Versi	Detail	Tanggal rilis
3.4.212	<ul style="list-style-type: none">• Perbaikan untuk EC2Config saat digunakan bersama dengan VM Import.• Memperbaiki masalah penamaan layanan di pemasang WiX.	
3.3.174	<ul style="list-style-type: none">• Penanganan pengecualian yang lebih baik untuk Systems Manager dan kegagalan penggabungan domain.• Ubah untuk mendukung versioning skema SSM Systems Manager.• Memperbaiki format disk sementara pada Win2K3.• Ubah untuk mendukung konfigurasi ukuran disk yang lebih besar dari 2TB.• Mengurangi penggunaan memori virtual dengan menyetel mode GC ke default.• Dukungan untuk mengunduh artefak dari jalur UNC di plugin <code>aws:psModule</code> dan <code>aws:application</code> .• Peningkatan logging untuk plugin aktivasi Windows.	

Versi	Detail	Tanggal rilis
3.2.97	<ul style="list-style-type: none">• Peningkatan performa dengan penundaan pemuatan rakitan Systems Manager SSM.• Penanganan pengecualian yang lebih baik untuk format sysprep2008.xml yang salah.• Dukungan baris perintah untuk konfigurasi "Terapkan" Systems Manager.• Ubah untuk mendukung penggabungan domain ketika ada penggantian nama komputer yang tertunda.• Dukungan untuk parameter opsional di plugin <code>aws:applications</code> .• Dukungan untuk array perintah di plugin <code>aws:psModule</code> .	
3.0.54	<ul style="list-style-type: none">• Aktifkan dukungan untuk Systems Manager.• Domain secara otomatis bergabung dengan instans EC2 Windows ke direktori AWS melalui Systems Manager.• Konfigurasi dan unggah CloudWatch log/metrik melalui Systems Manager.• Instal PowerShell modul melalui Systems Manager.• Instal aplikasi MSI melalui Systems Manager.	

Versi	Detail	Tanggal rilis
2.4.233	<ul style="list-style-type: none">• Menambahkan tugas terjadwal untuk memulihkan EC2Config dari kegagalan startup layanan.• Perbaiki pesan kesalahan log Konsol.• Pembaruan untuk meningkatkan dukungan untuk AWS layanan.	
2.3.313	<ul style="list-style-type: none">• Memperbaiki masalah dengan konsumsi memori yang besar dalam beberapa kasus ketika fitur CloudWatch Log diaktifkan.• Memperbaiki bug pemutakhiran sehingga ec2config versi yang lebih rendah dari 2.1.19 sekarang dapat dimutakhirkan ke versi terbaru.• Pengecualian pembukaan port COM yang diperbarui agar lebih ramah pengguna dan berguna dalam log.• configServiceSettings UI Ec2 menonaktifkan perubahan ukuran dan memperbaiki atribusi dan penempatan tampilan versi di UI.	
2.2.12	<ul style="list-style-type: none">• Ditangani NullPointerException saat menanyakan kunci registri untuk menentukan status Windows Sysprep yang mengembalikan null sesekali.• 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	<ul style="list-style-type: none">• Menghapus konfigurasi pengaturan CloudWatch Log melalui UI.• Memungkinkan pengguna untuk menentukan pengaturan CloudWatch Log dalam %ProgramFiles%\Amazon\Ec2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json file untuk memungkinkan perangkat tambahan di masa mendatang.	
2.2.9	Memperbaiki pengecualian yang tidak tertangani dan menambahkan logging.	
2.2.8	<ul style="list-style-type: none">• Memperbaiki pemeriksaan versi Windows OS in EC2Config Installer untuk mendukung Windows Server 2003 SP1 dan seterusnya.• Memperbaiki penanganan nilai null saat membaca kunci registri yang terkait dengan memperbarui file konfigurasi Sysprep.	
2.2.7	<ul style="list-style-type: none">• Menambahkan dukungan untuk EC2Config agar berjalan selama eksekusi Sysprep untuk Windows 2008 dan yang lebih tinggi.• Penanganan pengecualian dan logging yang lebih baik untuk diagnostik yang lebih baik	
2.2.6	<ul style="list-style-type: none">• Mengurangi beban pada instance dan pada CloudWatch Log saat mengunggah peristiwa log.• Mengatasi masalah pemutakhiran di mana plug-in CloudWatch Log tidak selalu diaktifkan	

Versi	Detail	Tanggal rilis
2.2.5	<ul style="list-style-type: none">• Menambahkan dukungan untuk mengunggah CloudWatch log ke Layanan Log.• Memperbaiki masalah syarat balapan di plug-in Ec2Output RDPcert• Mengubah opsi pemulihan Layanan EC2config untuk Restart dari Action TakeNo• Menambahkan lebih banyak informasi pengecualian saat EC2Config Rusak	
2.2.4	<ul style="list-style-type: none">• Memperbaiki kesalahan ketik di.cmd PostSysprep• Memperbaiki bug yang tidak disematkan otomatis oleh EC2Config ke menu awal untuk OS2012 +	

Versi	Detail	Tanggal rilis
2.2.3	<ul style="list-style-type: none"> • Menambahkan opsi untuk menginstal EC2Config tanpa langsung memulai layanan setelah instalasi. Untuk menggunakan, jalankan 'Ec2Install.exe start=false' dari prompt perintah • Menambahkan parameter di plugin wallpaper untuk mengontrol penambahan/penghapusan wallpaper. Untuk menggunakan, jalankan 'Ec2 WallpaperInfo .exe set' atau 'Ec2 .exe revert' dari command prompt WallpaperInfo • Ditambahkan memeriksa RealTimeUniversal kunci, menampilkan pengaturan yang salah dari kunci RealTimeUniversal registri ke Konsol • Menghapus ketergantungan EC2Config pada folder temp Windows • Dihapus ketergantungan UserData eksekusi pada .Net 3.5 	
2.2.2	<ul style="list-style-type: none"> • Menambahkan pemeriksaan ke perilaku penghentian layanan untuk memastikan bahwa sumber daya sedang dirilis • Memperbaiki masalah waktu eksekusi yang lama saat bergabung dengan domain 	
2.2.1	<ul style="list-style-type: none"> • Penginstal yang diperbarui untuk memungkinkan pemutakhiran dari versi sebelumnya • Memperbaiki WallpaperInfo bug Ec2 di lingkungan .Net4.5 saja • Memperbaiki bug deteksi driver yang terputus-putus • Menambahkan opsi instal diam. Menjalankan Ec2Install.exe dengan opsi '-q'. Misalnya: 'Ec2Install.exe -q' 	

Versi	Detail	Tanggal rilis
2.2.0	<ul style="list-style-type: none">• Menambahkan dukungan untuk lingkungan khusus .Net4 dan .Net4.5• Penginstal yang Diperbarui	
2.1.19	<ul style="list-style-type: none">• Menambahkan dukungan pelabelan disk sementara saat menggunakan driver jaringan Intel (mis. Tipe instans C3). Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di Amazon EC2.• Menambahkan Versi Asli AMI dan dukungan Nama Asli AMI ke output konsol• Membuat perubahan pada Output Konsol untuk pemformatan/ penguraian yang konsisten• File Bantuan yang Diperbarui	
2.1.18	<ul style="list-style-type: none">• Menambahkan Objek WMI EC2config untuk pemberitahuan Penyelesaian (-Namespace root\ Amazon -Class EC2_) ConfigService• Kueri Peningkatan Performa WMI Startup dengan Log Peristiwa besar; dapat menyebabkan CPU tinggi yang berkepanjangan selama eksekusi awal	

Versi	Detail	Tanggal rilis
2.1.17	<ul style="list-style-type: none">• Memperbaiki masalah UserData eksekusi dengan Output Standar dan pengisian buffer Kesalahan Standar• Sidik jari RDP yang salah yang terkadang muncul di Output Konsol untuk > = w2k8 OS telah diperbaiki• Output Konsol sekarang berisi 'RDPCERTIFICATE-SubjectName: 'untuk Windows 2008+, yang berisi nilai nama mesin• Menambahkan D:\ ke menu tarik-turun Pemetaan Huruf Drive• Tombol Bantuan dipindahkan ke kanan atas dan mengubah tampilan/nuansa• Menambahkan tautan survei Umpan Balik di kanan atas	

Versi	Detail	Tanggal rilis
2.1.16	<ul style="list-style-type: none">• Tab Umum menyertakan tautan ke halaman unduh EC2Config untuk Versi baru• Hamparan Wallpaper Desktop sekarang disimpan di folder Appdata Lokal Pengguna, bukan Dokumen Saya untuk mendukung pengalihan MyDoc• Nama MSSQLServer disinkronkan dengan sistem dalam skrip Post-Sysprep (2008+)• Folder Aplikasi yang Diurutkan Ulang (memindahkan file ke direktori Plugin dan menghapus file duplikat)• Output Log Sistem Diubah (Konsol):<ul style="list-style-type: none">* Telah memindahkan ke format tanggal, nama, nilai untuk penguraian yang lebih mudah (Harap mulai memigrasikan dependensi ke format baru)* Ditambahkan status plugin 'Ec2 SetPassword '* Telah menambahkan waktu Mulai dan Akhir Sysprep• Memperbaiki masalah Ephemeral Disks yang tidak diberi label sebagai 'Penyimpanan Sementara' untuk Sistem Operasi non-Inggris• Memperbaiki gagal Uninstall EC2Config setelah menjalankan Sysprep	

Versi	Detail	Tanggal rilis
2.1.15	<ul style="list-style-type: none">• Permintaan yang dioptimalkan ke layanan Metadata• Metadata sekarang melewati Pengaturan Proxy• Ephemeral Disks diberi label sebagai 'Penyimpanan Sementara' dan Important.txt ditempatkan pada volume saat ditemukan (hanya driver Citrix PV). Untuk informasi selengkapnya, lihat Mutakhirkan driver PV pada instans Windows.• 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)• UserData sekarang berjalan segera setelah 'Windows Siap'	
2.1.14	Perbaikan wallpaper desktop	
2.1.13	<ul style="list-style-type: none">• Wallpaper desktop akan menampilkan nama host secara default• Ketergantungan yang dihapus pada layanan Windows Time• Rute ditambahkan jika ada beberapa IP ditugaskan ke satu antarmuka	
2.1.11	<ul style="list-style-type: none">• Perubahan dilakukan pada Plugin Ec2Activation• -Memverifikasi status Aktivasi setiap 30 hari• -Jika Masa Tenggang memiliki sisa 90 hari (dari 180 hari), coba kembali aktivasi	

Versi	Detail	Tanggal rilis
2.1.10	<ul style="list-style-type: none">• Hamparan wallpaper desktop tidak lagi dipertahankan dengan Sysprep atau Mati tanpa Sysprep• Opsi data pengguna untuk dijalankan pada setiap layanan dimulai dengan <code><persist>>true</persist></code>• Mengubah lokasi dan nama/DisableWinupdate.cmd menjadi /Script/ .cmd PostSysprep• Kata sandi administrator diatur agar tidak kedaluwarsa secara default di PostSysprep /Script/ .cmd• Uninstall akan menghapus PostSysprep skrip EC2config dari <code>c:\windows\setup\script\ .cmd CommandComplete</code>• Tambah Rute mendukung metrik antarmuka kustom	
2.1.9	UserData Eksekusi tidak lagi terbatas pada 3851 Karakter	

Versi	Detail	Tanggal rilis
2.1.7	<ul style="list-style-type: none">• Versi OS dan pengenalan bahasa yang ditulis pada konsol• Versi EC2Config yang ditulis pada konsol• Versi driver PV yang ditulis pada konsol• Deteksi Pemeriksaan Bug dan output ke konsol pada boot berikutnya ketika ditemukan• Opsi ditambahkan ke config.xml untuk mempertahankan kredensial Sysprep• Tambahkan logika Coba Ulang Rute jika ENI tidak tersedia di awal• PID eksekusi Data Pengguna ditulis ke konsol• Panjang kata sandi minimum yang dihasilkan diambil dari GPO• Setelah layanan mulai mencoba lagi 3 kali• Menambahkan contoh file DownloadFile S3_.ps1 dan S3_Upload file.ps1 ke folder/Scripts	

Versi	Detail	Tanggal rilis
2.1.6	<ul style="list-style-type: none">• Informasi versi ditambahkan ke tab Umum• Mengganti nama tab Bundel menjadi Gambar• Menyederhanakan proses menentukan kata sandi dan memindahkan UI terkait kata sandi dari tab Umum ke tab Gambar• Mengganti nama tab Pengaturan Disk menjadi Penyimpanan• Menambahkan tab Dukungan dengan alat umum untuk pemecahan masalah• <code>sysprep.ini</code> Windows Server 2003 diatur untuk memperluas partisi OS secara default• Menambahkan alamat IP privat ke wallpaper• Alamat IP pribadi ditampilkan di wallpaper• Menambahkan logika coba lagi untuk output Konsol• Memperbaiki pengecualian Com port untuk aksesibilitas metadata – menyebabkan EC2Config berakhir sebelum output konsol ditampilkan• Memeriksa status aktivasi pada setiap boot -- aktifkan seperlunya• Memperbaiki masalah jalur relatif -- yang disebabkan saat eksekusi pintasan wallpaper secara manual dari folder startup; menunjuk ke Administrator/log• Memperbaiki warna latar belakang default untuk pengguna Windows Server 2003 (selain Administrator)	

Versi	Detail	Tanggal rilis
2.1.2	<ul style="list-style-type: none">• Stempel waktu konsol dalam UTC (Zulu)• Tampilan hyperlink pada tab Sysprep dihapus• Penambahan fitur untuk memperluas Volume Root secara dinamis saat boot pertama untuk Windows 2008+• Ketika Set-Password diaktifkan, sekarang secara otomatis EC2Config dapat mengatur kata sandi• EC2Config memeriksa status aktivasi sebelum menjalankan Sysprep (menampilkan peringatan jika tidak diaktifkan)• Sysprep.xml Windows Server 2003 sekarang default ke zona waktu UTC, bukan Pasifik• Server Aktivasi Acak• Mengganti nama tab Pemetaan Drive menjadi Pengaturan Disk• Pindah Inisialisasi item UI Drive dari Umum ke tab Pengaturan Disk• Tombol bantuan sekarang mengarah ke file bantuan HTML• File bantuan HTML yang diperbarui dengan perubahan• 'Catatan' yang diperbarui untuk Pemetaan Huruf Drive• Ditambahkan InstallUpdates.ps1 ke/Scripts folder untuk mengotomatisasi Patch dan pembersihan sebelum Sysprep	

Versi	Detail	Tanggal rilis
2.1.0	<ul style="list-style-type: none">Wallpaper desktop menampilkan informasi instans secara default saat logon pertama kali (tidak memutuskan/menghubungkan kembali)PowerShell dapat dijalankan dari data pengguna dengan mengelilingi kode dengan <code><powershell></powershell></code>	

Berlangganan notifikasi layanan EC2Config

Amazon SNS dapat memberi Anda notifikasi saat layanan EC2Config versi baru sedang dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Untuk 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 ARN Topik, salin Amazon Resource Name (ARN) berikut:

```
arn:aws:sns:us-east-1:801119661308:ec2-windows-ec2config
```
 - 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 yang meminta Anda untuk mengonfirmasi langganan Anda. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Kapan pun versi baru dari layanan EC2Config dirilis, kami akan mengirim notifikasi ke pelanggan. Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Untuk berhenti berlangganan notifikasi EC2Config

1. Buka konsol Amazon SNS.
2. Di panel navigasi, pilih Langganan.
3. Pilih langganan lalu pilih Tindakan, Hapus langganan Saat diminta konfirmasi, pilih Hapus.

Pemecahan masalah layanan 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 memperbaharui EC2Config pada instans Windows yang didukung Amazon EBS yang tidak dapat Anda hubungkan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Contoh.
3. Cari instans yang terpengaruh. Pilih instans dan pilih status Instans, lalu pilih Hentikan instans.

 Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan data, pastikan untuk mencadangkannya 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 keTemp 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. Memilih HKEY_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. [Hentikan dan mulai 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 String Value yang Anda buat sebelumnya: `AutoAdminLogon`, `DefaultUserName`, dan `DefaultPassword`.
23. Hapus atau hentikan instans sementara yang Anda buat dalam prosedur ini.

Gunakan EC2 Fast Launch untuk instans Windows Anda

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. Amazon EC2 Windows Server AMI yang mengaktifkan fitur Peluncuran Cepat EC2, selesaikan beberapa langkah tersebut dan reboot terlebih dahulu untuk mengurangi waktu yang diperlukan untuk meluncurkan instans.

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 yang 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 [Pertimbangan](#). 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 sumber daya dengan EC2 Fast Launch](#).

Daftar Isi

- [Istilah kunci](#)
- [Prasyarat Peluncuran Cepat EC2](#)
- [Konfigurasi pengaturan Peluncuran Cepat EC2 untuk AMI Amazon EC2 Windows Server Anda](#)
- [Lihat AMI dengan Peluncuran Cepat EC2 diaktifkan](#)
- [Kelola biaya sumber daya dengan EC2 Fast Launch](#)
- [Pantau Peluncuran Cepat EC2](#)
- [Peran terkait layanan untuk Peluncuran Cepat 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.

Prasyarat Peluncuran Cepat EC2

Sebelum menyiapkan Peluncuran Cepat EC2, verifikasi bahwa Anda telah memenuhi prasyarat berikut yang diperlukan untuk membuat snapshot untuk AMI di: Akun AWS

- Jika Anda tidak menggunakan template peluncuran untuk mengonfigurasi pengaturan, pastikan VPC default dikonfigurasi untuk Wilayah tempat Anda menggunakan Peluncuran Cepat EC2.

Note

Jika Anda secara tidak sengaja menghapus VPC default di Wilayah tempat Anda berencana untuk mengonfigurasi Peluncuran Cepat EC2, Anda dapat membuat VPC default baru di Wilayah tersebut. Untuk mempelajari selengkapnya, lihat [Membuat VPC default](#) di Panduan Pengguna Amazon VPC.

- Untuk menentukan VPC non-default, Anda harus menggunakan templat peluncuran saat Anda mengonfigurasi peluncuran cepat Windows. Untuk informasi selengkapnya, lihat [Gunakan template peluncuran saat Anda mengatur Peluncuran Cepat EC2](#).
- Jika akun Anda memiliki kebijakan yang memberlakukan IMDSv2 untuk instans Amazon EC2, Anda harus membuat templat peluncuran yang menentukan konfigurasi metadata untuk menerapkan ImDSv2.
- AMI Peluncuran Cepat EC2 Pribadi harus mendukung eksekusi skrip data pengguna.
- Untuk mengonfigurasi Peluncuran Cepat EC2 untuk AMI, Anda harus membuat AMI menggunakan Sysprep opsi shutdown. Fitur Peluncuran Cepat EC2 saat ini tidak mendukung AMI yang dibuat dari instance yang sedang berjalan.

Untuk membuat AMI menggunakan Sysprep, lihat [Buat AMI dengan Windows Sysprep](#).

- Kuota default untuk Maksimal peluncuran paralel di semua AMI pada Akun AWS adalah 40 per Wilayah. Anda dapat meminta peningkatan Kuota Layanan untuk akun Anda, sebagai berikut.
 1. [Masuk ke AWS Management Console dan buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/](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 tautan untuk Parallel instance launches. Proses ini akan membawa Anda ke halaman detail kuota layanan Peluncuran instans paralel.
 5. Pilih Ajukan peningkatan kuota.

Untuk informasi lebih lanjut, lihat [Meminta peningkatan kuota](#) di Panduan Pengguna Kuota Layanan.

Konfigurasi pengaturan Peluncuran Cepat EC2 untuk AMI Amazon EC2 Windows Server Anda

Anda dapat mengonfigurasi EC2 Fast Launch untuk AMI Windows yang Anda miliki, atau AMI yang dibagikan dengan Anda dari AWS Management Console, API, SDK CloudFormation, atau AWS Command Line Interface (AWS CLI). Sebelum Anda mengonfigurasi Peluncuran Cepat EC2, verifikasi bahwa AMI Anda memenuhi semua prasyarat yang diperlukan untuk membuat snapshot yang telah disediakan sebelumnya. Untuk informasi selengkapnya, lihat [Prasyarat Peluncuran Cepat EC2](#).

Bagian berikut mencakup langkah-langkah konfigurasi untuk konsol Amazon EC2 dan AWS CLI

Aktifkan Peluncuran Cepat EC2

Untuk mengaktifkan Peluncuran Cepat EC2, pilih tab yang cocok dengan lingkungan Anda, dan ikuti langkah-langkahnya.

Note

Sebelum mengubah pengaturan ini, pastikan AMI Anda, dan Wilayah yang Anda jalankan memenuhi semua [Prasyarat Peluncuran Cepat EC2](#).

Console

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 Konfigurasi peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk Peluncuran Cepat EC2.
5. Untuk mulai menggunakan snapshot yang telah tersedia untuk meluncurkan instans dari AMI Windows Anda secara 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.

- Setelah selesai membuat perubahan, pilih Simpan perubahan.

Note

Jika Anda perlu menggunakan templat peluncuran untuk menentukan VPC non-default, atau untuk mengonfigurasi pengaturan metadata untuk IMDSv2, lihat [Gunakan template peluncuran saat Anda mengatur Peluncuran Cepat EC2](#).

AWS CLI

enable-fast-launch Perintah tersebut memanggil operasi API [EnableFastPeluncuran](#) Amazon EC2.

Sintaksis:

```
aws ec2 enable-fast-launch \  
  --image-id <value> \  
  --resource-type <value> \ (optional)  
  --snapshot-configuration <value> \ (optional)  
  --launch-template <value> \ (optional)  
  --max-parallel-launches <value> \ (optional)  
  --dry-run | --no-dry-run \ (optional)  
  --cli-input-json <value> \ (optional)  
  --generate-cli-skeleton <value> \ (optional)
```

Contoh:

Contoh [enable-fast-launch](#) berikut memungkinkan Peluncuran Cepat EC2 untuk AMI yang ditentukan, meluncurkan enam instance paralel untuk pra-penyediaan. ResourceType diatur ke snapshot, yang merupakan nilai default.

```
aws ec2 enable-fast-launch \  
  --image-id ami-01234567890abcdef \  
  --max-parallel-launches 6 \  
  --resource-type snapshot
```

Output:

```
{
```

```

    "ImageId": "ami-01234567890abcdef",
    "ResourceType": "snapshot",
    "SnapshotConfiguration": {
        "TargetResourceCount": 10
    },
    "LaunchTemplate": {},
    "MaxParallelLaunches": 6,
    "OwnerId": "0123456789123",
    "State": "enabling",
    "StateTransitionReason": "Client.UserInitiated",
    "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}

```

Tools for PowerShell

Enable-EC2FastLaunchCmdlet memanggil operasi Amazon [EnableFastEC2](#) Launch API untuk mengaktifkan Peluncuran Cepat EC2 di AMI Windows Anda.

Sintaksis:

```

Enable-EC2FastLaunch
  -ImageId <String>
  -LaunchTemplate_LaunchTemplateId <String>
  -LaunchTemplate_LaunchTemplateName <String>
  -MaxParallelLaunch <Int32>
  -ResourceType <String>
  -SnapshotConfiguration_TargetResourceCount <Int32>
  -LaunchTemplate_Version <String>
  -Select <String>
  -PassThru <SwitchParameter>
  -Force <SwitchParameter>

```

Contoh:

[Enable-EC2FastLaunch](#) Contoh berikut memungkinkan Peluncuran Cepat EC2 untuk AMI yang ditentukan, meluncurkan enam instance paralel untuk pra-penyediaan. ResourceType diatur ke snapshot, yang merupakan nilai default.

```

Enable-EC2FastLaunch `
  -ImageId ami-01234567890abcdef `
  -MaxParallelLaunch 6 `
  -Region us-west-2 `

```

-ResourceType *snapshot*

Output:

```
ImageId           : ami-01234567890abcdef
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

Untuk menonaktifkan Peluncuran Cepat EC2, pilih tab yang cocok dengan lingkungan Anda, dan ikuti langkah-langkahnya.

Note

Sebelum mengubah pengaturan ini, pastikan AMI Anda, dan Wilayah yang Anda jalankan memenuhi semua [Prasyarat Peluncuran Cepat EC2](#).

Console

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 Konfigurasi peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk Peluncuran Cepat EC2.
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

disable-fast-launch Perintah tersebut memanggil operasi API [DisableFastPeluncuran](#) Amazon EC2.

Sintaksis:

```
aws ec2 disable-fast-launch \  
  --image-id <value> \  
  --force | --no-force \ (optional)  
  --dry-run | --no-dry-run \ (optional)  
  --cli-input-json <value> \ (optional)  
  --generate-cli-skeleton <value> \ (optional)
```

Contoh:

Contoh [disable-fast-launch](#) berikut menonaktifkan Peluncuran Cepat EC2 pada AMI yang ditentukan, dan membersihkan snapshot yang sudah disediakan sebelumnya.

```
aws ec2 disable-fast-launch \  
  --image-id ami-01234567890abcdef
```

Output:

```
{  
  "ImageId": "ami-01234567890abcdef",  
  "ResourceType": "snapshot",  
  "SnapshotConfiguration": {},  
  "LaunchTemplate": {  
    "LaunchTemplateId": "lt-01234567890abcdef",  
    "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"
}
```

Tools for PowerShell

Disable-EC2FastLaunchCmdlet memanggil operasi API Peluncuran Amazon [DisableFastEC2](#).

Sintaksis:

```
Disable-EC2FastLaunch
  -ImageId <String>
  -ForceStop <Boolean>
  -Select <String>
  -PassThru <SwitchParameter>
  -Force <SwitchParameter>
```

Contoh:

[Disable-EC2FastLaunch](#) Contoh berikut menonaktifkan Peluncuran Cepat EC2 pada AMI yang ditentukan, dan membersihkan snapshot yang sudah ada sebelumnya.

```
Disable-EC2FastLaunch -ImageId ami-01234567890abcdef
```

Output:

```
ImageId           : ami-01234567890abcdef
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 Peluncuran Cepat EC2

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 sebuah templat peluncuran untuk menentukan VPC non-default untuk AMI Windows Anda.
- 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 panggil tindakan `Launch API.EnableFast`](#)

Amazon EC2 EC2 Fast Launch tidak mendukung konfigurasi berikut saat Anda menggunakan template peluncuran. Jika Anda menggunakan template peluncuran untuk Peluncuran Cepat EC2, 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 instance spot

Menentukan VPC non-default

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 [Membuat templat peluncuran](#).

Langkah 2: Tentukan template peluncuran untuk AMI Peluncuran Cepat EC2 Anda

Pilih tab yang cocok dengan proses Anda:

Console

Untuk menentukan template peluncuran EC2 Fast Launch dari AWS Management Console, ikuti langkah-langkah berikut:

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 Konfigurasi peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk Peluncuran Cepat EC2.
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, API

Untuk menentukan template peluncuran untuk EC2 Fast Launch dari AWS CLI, tentukan nama template peluncuran atau ID di `--launch-template` parameter saat Anda menjalankan perintah [enable-fast-launch](#) di AWS CLI

Untuk menentukan template peluncuran EC2 Fast Launch dalam permintaan API, tentukan nama atau ID template peluncuran di `LaunchTemplate` parameter saat Anda memanggil tindakan [EnableFastLaunch](#) API.

Untuk informasi selengkapnya tentang templat peluncuran EC2, lihat [Meluncurkan sebuah instans dari templat peluncuran](#).

Buat gambar khusus dengan EC2 Fast Launch diaktifkan

Amazon EC2 EC2 Fast Launch terintegrasi dengan EC2 Image Builder untuk membantu Anda membuat gambar khusus dengan Peluncuran Cepat EC2 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.

Lihat AMI dengan Peluncuran Cepat EC2 diaktifkan

Anda dapat menggunakan perintah [describe-fast-launch-images](#) di AWS CLI, atau [Get-EC2FastLaunchImage](#) Tools for PowerShell Cmdlet untuk mendapatkan detail untuk AMI yang mengaktifkan EC2 Fast Launch.

Amazon EC2 memberikan detail berikut untuk setiap AMI Windows yang muncul dalam hasil:

- ID gambar untuk AMI dengan Peluncuran Cepat EC2 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 Peluncuran Cepat EC2 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 Peluncuran Cepat EC2 untuk AMI terkait berubah ke status saat ini.
- Waktu Peluncuran Cepat EC2 untuk AMI terkait berubah ke status saat ini.

Pilih tab yang cocok dengan lingkungan baris perintah Anda:

AWS CLI

`describe-fast-launch-images` Perintah tersebut memanggil operasi Amazon EC2 [DescribeFastLaunchImages](#) API.

Sintaksis:

```
aws ec2 describe-fast-launch-images \
  --image-ids <value> \ (optional)
  --filters <value> \ (optional)
  --dry-run | --no-dry-run \ (optional)
  --cli-input-json <value> \ (optional)
  --starting-token <value> \ (optional)
  --page-size <value> \ (optional)
  --max-items <value> \ (optional)
  --generate-cli-skeleton <value> \ (optional)
```

Contoh:

Contoh [deskripsi-fast-launch-images](#) berikut menjelaskan detail untuk masing-masing AMI di akun yang dikonfigurasi untuk Peluncuran Cepat EC2. Dalam contoh ini, hanya satu AMI di akun yang dikonfigurasi untuk Peluncuran Cepat EC2.

```
aws ec2 describe-fast-launch-images
```

Output:

```
{
  "FastLaunchImages": [
    {
      "ImageId": "ami-01234567890abcdef",
      "ResourceType": "snapshot",
      "SnapshotConfiguration": {},
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-01234567890abcdef",
        "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"
    }
  ]
}
```

Tools for PowerShell

Get-EC2FastLaunchImageCmdlet memanggil operasi Amazon [DescribeFastLaunchImages](#) EC2 API.

Sintaksis:

```
Get-EC2FastLaunchImage
-Filter <Filter[]>
-ImageId <String[]>
-MaxResult <Int32>
-NextToken <String>
-Select <String>
-NoAutoIteration <SwitchParameter>
```

Contoh:

[Get-EC2FastLaunchImage](#) Contoh berikut menjelaskan detail untuk masing-masing AMI di akun yang dikonfigurasi untuk Peluncuran Cepat EC2. Dalam contoh ini, hanya satu AMI di akun yang dikonfigurasi untuk Peluncuran Cepat EC2.

```
Get-EC2FastLaunchImage -ImageId ami-01234567890abcdef
```

Output:

```
ImageId           : ami-01234567890abcdef
LaunchTemplate    :
  Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches : 6
OwnerId          : 0123456789123
ResourceType     : snapshot
SnapshotConfiguration :
State            : enabled
StateTransitionReason : Client.UserInitiated
StateTransitionTime  : 2/25/2022 12:54:43 PM
```

Kelola biaya sumber daya dengan EC2 Fast Launch

Tidak ada biaya layanan untuk mengkonfigurasi AMI Windows untuk Peluncuran Cepat EC2. Namun, saat Anda mengaktifkan Peluncuran Cepat EC2 untuk Amazon EC2 Windows AMI, harga standar berlaku untuk sumber daya AWS dasar yang digunakan Amazon EC2 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 Cepat EC2. Untuk informasi selengkapnya tentang cara mengonfigurasi tag alokasi biaya, lihat [Lacak biaya Peluncuran Cepat EC2 pada tagihan Anda](#).

Contoh berikut menunjukkan bagaimana biaya yang terkait dengan biaya snapshot Peluncuran Cepat EC2 mungkin dialokasikan.

Contoh skenario: Perusahaan AtoZ memiliki AMI Windows dengan volume root 50 GiB EBS. Mereka mengaktifkan Peluncuran Cepat EC2 untuk AMI mereka, dan menetapkan jumlah sumber daya target menjadi lima. Selama sebulan, menggunakan EC2 Fast Launch untuk AMI mereka harganya sekitar \$5,00, dan rincian biaya adalah sebagai berikut:

1. Saat Contoh AtoZ mengaktifkan Peluncuran Cepat EC2, 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, Amazon EC2 mengambil snapshot instans, 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, ia menghentikan instans. 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 Cepat EC2 pada tagihan Anda

Tag alokasi biaya dapat membantu Anda mengatur AWS tagihan untuk mencerminkan biaya yang terkait dengan Peluncuran Cepat EC2. Anda dapat menggunakan tag berikut yang ditambahkan Amazon EC2 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 mengaktifkan 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 mengaktifkan tag tersebut terlebih dahulu di konsol Manajemen Penagihan dan Biaya. Untuk informasi selengkapnya, lihat [Mengaktifkan Tag Alokasi Biaya Buatan Pengguna](#) dalam Panduan Pengguna AWS Billing and Cost Management .

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 Billing and Cost Management .
- 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 EC2 Fast Launch untuk AMI Windows yang Anda miliki, atau AMI yang dibagikan dengan Anda dari konsol Amazon EC2, API, SDK [CloudFormation](#), ec2 atau perintah di AWS CLI Bagian berikut mencakup langkah-langkah konfigurasi untuk konsol Amazon EC2 dan AWS CLI

Anda juga dapat membuat AMI Windows kustom yang dikonfigurasi untuk EC2 Fast Launch dengan EC2 Image Builder. Untuk informasi selengkapnya, lihat [Membuat setelan distribusi untuk Windows AMI dengan EC2 Fast Launch enabled \(AWS CLI\)](#).

Pantau Peluncuran Cepat EC2

Bagian ini mencakup cara memantau AMI Amazon EC2 Windows Server di akun Anda yang mengaktifkan Peluncuran Cepat EC2.

Pantau perubahan status Peluncuran Cepat EC2 dengan EventBridge

Saat status berubah untuk AMI Windows dengan Peluncuran Cepat EC2 diaktifkan, Amazon EC2 menghasilkan EC2 Fast Launch State-change Notification peristiwa. Kemudian Amazon EC2 mengirimkan peristiwa perubahan status ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon Events). CloudWatch

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 Cepat EC2 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 Cepat EC2 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 Cepat EC2.

resourceType

Tipe sumber daya yang digunakan untuk pra-penyediaan. Nilai yang didukung: snapshot. Nilai default-nya adalah snapshot.

status

Status saat ini dari fitur Peluncuran Cepat EC2 untuk AMI yang ditentukan. Nilai-nilai yang valid meliputi:

- **mengaktifkan** - Anda telah mengaktifkan fitur Peluncuran Cepat EC2 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 Cepat EC2 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 Peluncuran Cepat EC2 Anda melewati proses pra-penyediaan. Ini dapat terjadi jika fitur Peluncuran Cepat EC2 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 Cepat EC2 untuk AMI, dan Amazon EC2 telah memulai proses pembersihan.
- **dinonaktifkan** - Fitur Peluncuran Cepat EC2 dinonaktifkan. Status berubah menjadi `disabled` sesaat setelah Amazon EC2 menyelesaikan proses pembersihan.
- **gagal-memonaktifkan** - Terjadi kesalahan yang menyebabkan proses pembersihan gagal. Ini berarti bahwa beberapa snapshot pra-penyediaan mungkin masih ada di akun.

negara TransitionReason

Alasan bahwa keadaan berubah untuk EC2 Fast Launch AMI.

Note

Semua bidang dalam pesan peristiwa ini diperlukan.

Contoh berikut menunjukkan AMI Peluncuran Cepat EC2 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 Cepat EC2 dengan CloudWatch

AMI Amazon EC2 dengan Peluncuran Cepat EC2 mengaktifkan metrik kirim ke Amazon CloudWatch. Anda dapat menggunakan API AWS Management Console AWS CLI, atau API untuk membuat daftar metrik yang dikirimkan oleh Peluncuran Cepat EC2. CloudWatch AWS/EC2Namespace mencakup metrik Peluncuran Cepat EC2 berikut:

Metrik	Deskripsi
NumberOfAvailableFastLaunchSnapshots	Jumlah snapshot pra-penyediaan yang tersedia per Peluncuran Cepat EC2 mengaktifkan AMI.
NumberOfInstancesFastDiluncurkan	Jumlah instans per Peluncuran Cepat EC2 memungkinkan AMI yang diluncurkan dari snapshot yang telah disediakan sebelumnya.
NumberOfInstancesNotFastLaunched	Jumlah instans per Peluncuran Cepat EC2 memungkinkan AMI yang menghasilkan boot dingin karena kurangnya snapshot pra-penyediaan yang tersedia pada waktu peluncuran.

Metrik	Deskripsi
FastLaunchSnapshotUsedToRefillStartTime	Stempel waktu saat Amazon EC2 meluncurkan gambar baru dari Peluncuran Cepat EC2 memungkinkan AMI membuat snapshot lain setelah snapshot yang ada digunakan.
FastLaunchSnapshotCreationWaktu	Mengukur waktu yang dibutuhkan Amazon EC2 untuk meluncurkan instans dan membuat snapshot untuk AMI yang mengaktifkan Peluncuran Cepat EC2.

Peran terkait layanan 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 unik peran IAM yang ditautkan langsung ke sebuah Layanan AWS. Peran terkait layanan 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 paket snapshot pra-penyediaan yang mengurangi waktu yang diperlukan untuk meluncurkan instans dari AMI Windows Anda.

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda mulai menggunakan Peluncuran Cepat EC2 untuk AMI, Amazon EC2 membuat peran terkait layanan untuk Anda, jika belum ada.

Note

Jika peran terkait layanan dihapus dari akun Anda, Anda dapat mengaktifkan Peluncuran Cepat EC2 untuk AMI Windows lain untuk membuat ulang peran di akun Anda. Atau, Anda dapat menonaktifkan Peluncuran Cepat EC2 untuk AMI Anda saat ini, dan kemudian 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 kembali penggunaan EC2 Fast Launch untuk AMI Anda.

Amazon EC2 tidak mengizinkan Anda untuk mengubah peran tertaut layanan `AWSServiceRoleForEC2FastLaunch`. Setelah Anda membuat peran tertaut 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 lebih lanjut, lihat [Mengubah Peran Tertaut Layanan](#) di Panduan Pengguna IAM.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus semua sumber daya terkait terlebih dahulu. Ini melindungi sumber daya Amazon EC2 yang terkait dengan Amazon EC2 Windows Server AMI Anda dengan Peluncuran Cepat EC2 diaktifkan, karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Amazon EC2 mendukung peran terkait layanan Peluncuran Cepat EC2 di semua Wilayah tempat layanan Amazon EC2 tersedia. Untuk informasi selengkapnya, lihat [Wilayah](#).

Izin diberikan oleh **`AWSServiceRoleForEC2FastLaunch`**

Amazon EC2 menggunakan kebijakan yang dikelola `EC2FastLaunchServiceRolePolicy` untuk menyelesaikan tindakan berikut:

- `cloudwatch:PutMetricData`— Posting data metrik yang terkait dengan Peluncuran Cepat EC2 ke namespace Amazon EC2.
- `ec2:CreateLaunchTemplate`— Buat template peluncuran untuk Amazon EC2 Windows Server AMI Anda dengan Peluncuran Cepat EC2 diaktifkan.
- `ec2:CreateSnapshot`— Buat snapshot yang telah disediakan sebelumnya untuk AMI Amazon EC2 Windows Server Anda dengan Peluncuran Cepat EC2 diaktifkan.
- `ec2:CreateTags`— Buat tag untuk sumber daya yang terkait dengan peluncuran dan pra-penyediaan instans Windows untuk Amazon EC2 Windows Server AMI Anda dengan Peluncuran Cepat EC2 diaktifkan.
- `ec2:DeleteSnapshots`— Hapus semua snapshot yang telah disediakan sebelumnya jika Peluncuran Cepat EC2 dimatikan untuk AMI yang diaktifkan sebelumnya.
- `ec2:DescribeImages` – Menjelaskan gambar untuk semua sumber daya.
- `ec2:DescribeInstanceAttribute` – Menjelaskan atribut instans untuk semua sumber daya.

- `ec2:DescribeInstanceStatus` – Menjelaskan status instans untuk semua sumber daya.
- `ec2:DescribeInstances` – Menjelaskan instans untuk semua sumber daya.
- `ec2:DescribeInstanceTypeOfferings` – Menjelaskan penawaran tipe instans untuk semua sumber daya.
- `ec2:DescribeLaunchTemplates` – Menjelaskan templat peluncuran untuk semua sumber daya.
- `ec2:DescribeLaunchTemplateVersions` – Menjelaskan versi templat peluncuran untuk semua sumber daya.
- `ec2:DescribeSnapshots` – Menjelaskan sumber daya snapshot untuk semua sumber daya.
- `ec2:DescribeSubnets` – Menjelaskan subnet untuk semua sumber daya.
- `ec2:RunInstances`— Luncurkan instans dari Amazon EC2 Windows Server AMI dengan Peluncuran Cepat EC2 diaktifkan, untuk melakukan langkah-langkah penyediaan.
- `ec2:StopInstances`— Hentikan instans yang diluncurkan dari Amazon EC2 Windows Server AMI dengan Peluncuran Cepat EC2 diaktifkan, untuk membuat snapshot yang telah disediakan sebelumnya.
- `ec2:TerminateInstances`- Hentikan instance yang diluncurkan dari Amazon EC2 Windows Server AMI dengan Peluncuran Cepat EC2 diaktifkan, setelah membuat snapshot yang telah disediakan sebelumnya darinya.
- `iam:PassRole` – Memungkinkan peran tertaut layanan `AWSServiceRoleForEC2FastLaunch` untuk meluncurkan instans mewakili Anda menggunakan profil instans dari templat peluncuran Anda.

Untuk informasi selengkapnya tentang penggunaan kebijakan terkelola untuk Amazon EC2, lihat [AWS kebijakan terkelola untuk Amazon EC2](#).

Akses ke kunci terkelola pelanggan untuk digunakan dengan AMI terenkripsi dan snapshot EBS

Prasyarat

- Untuk memungkinkan Amazon EC2 mengakses AMI terenkripsi mewakili Anda, Anda harus memiliki izin untuk tindakan `createGrant` dalam kunci yang dikelola pelanggan.

Saat Anda mengaktifkan Peluncuran Cepat EC2 untuk AMI terenkripsi, Amazon EC2 memastikan bahwa izin diberikan untuk peran `AWSServiceRoleForEC2FastLaunch` tersebut menggunakan kunci yang dikelola pelanggan untuk mengakses AMI Anda. Izin ini diperlukan untuk meluncurkan instans dan membuat snapshot pra-penyediaan atas nama Anda.

Gunakan akselerator Amazon Elastic Graphics pada instans Windows

Important

Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.

Amazon Elastic Graphics memberikan akselerasi grafis yang fleksibel, berbiaya rendah, dan berperforma tinggi untuk instans Windows Anda. Akselerator Elastic Graphics hadir dalam berbagai ukuran dan merupakan alternatif berbiaya rendah untuk menggunakan tipe instance grafis GPU (seperti G3). Anda memiliki fleksibilitas untuk memilih tipe instans yang memenuhi kebutuhan komputasi, memori, dan penyimpanan aplikasi Anda. Kemudian, pilih akselerator untuk instans Anda yang memenuhi persyaratan grafis beban kerja Anda.

Elastic Graphics cocok untuk aplikasi yang memerlukan akselerator grafis tambahan dalam jumlah kecil atau intermiten, dan yang menggunakan dukungan grafik OpenGL. Jika Anda membutuhkan akses ke GPU lengkap yang langsung dilampirkan, dan penggunaan kerangka kerja komputasi paralel DirectX, CUDA, atau Open Computing Language (OpenCL), gunakan instans tipe instans komputasi terakselerasi sebagai gantinya.

Daftar Isi

- [Dasar-dasar Elastic Graphics](#)
- [Harga untuk Elastic Graphics](#)
- [Batasan Elastic Graphics](#)
- [Bekerja dengan Elastic Graphics](#)
- [Pemeliharaan Elastic Graphics](#)
- [Gunakan CloudWatch metrik untuk memantau Grafik Elastis](#)
- [Pemecahan Masalah](#)

Dasar-dasar Elastic Graphics

Untuk menggunakan Elastic Graphics, luncurkan instance Windows dan tentukan jenis akselerator untuk instance selama peluncuran. AWS menemukan kapasitas Elastic Graphics yang tersedia dan membuat koneksi jaringan antara instans Anda dan akselerator Elastic Graphics.

Note

Instans bare metal tidak didukung.

Akselerator Elastic Graphics tersedia di AWS Wilayah berikut: `us-east-1`, `us-east-2`, `us-west-2`, `ap-northeast-1`, `ap-southeast-1`, `ap-southeast-2`, `eu-central-1`, dan `eu-west-1`.

Tipe instans berikut mendukung akselerator Elastic Graphics:

- Tujuan umum: M3, M4, M5, M5d, M5dn, M5n, T2, T3

Note

Hanya `t2.medium` dan lebih besar serta `t3.medium` dan lebih besar yang didukung.

- Komputasi yang dioptimalkan: C3, C4, C5, C5a, C5ad, C5d, C5n
- Memori yang dioptimalkan: R3, R4, R5, R5d, R5dn, R5n, X1, X1e, z1d
- Penyimpanan yang dioptimalkan: D2, D3, D3en, H1, I3, I3en
- Komputasi terakselerasi: P2, P3, dan P3dn

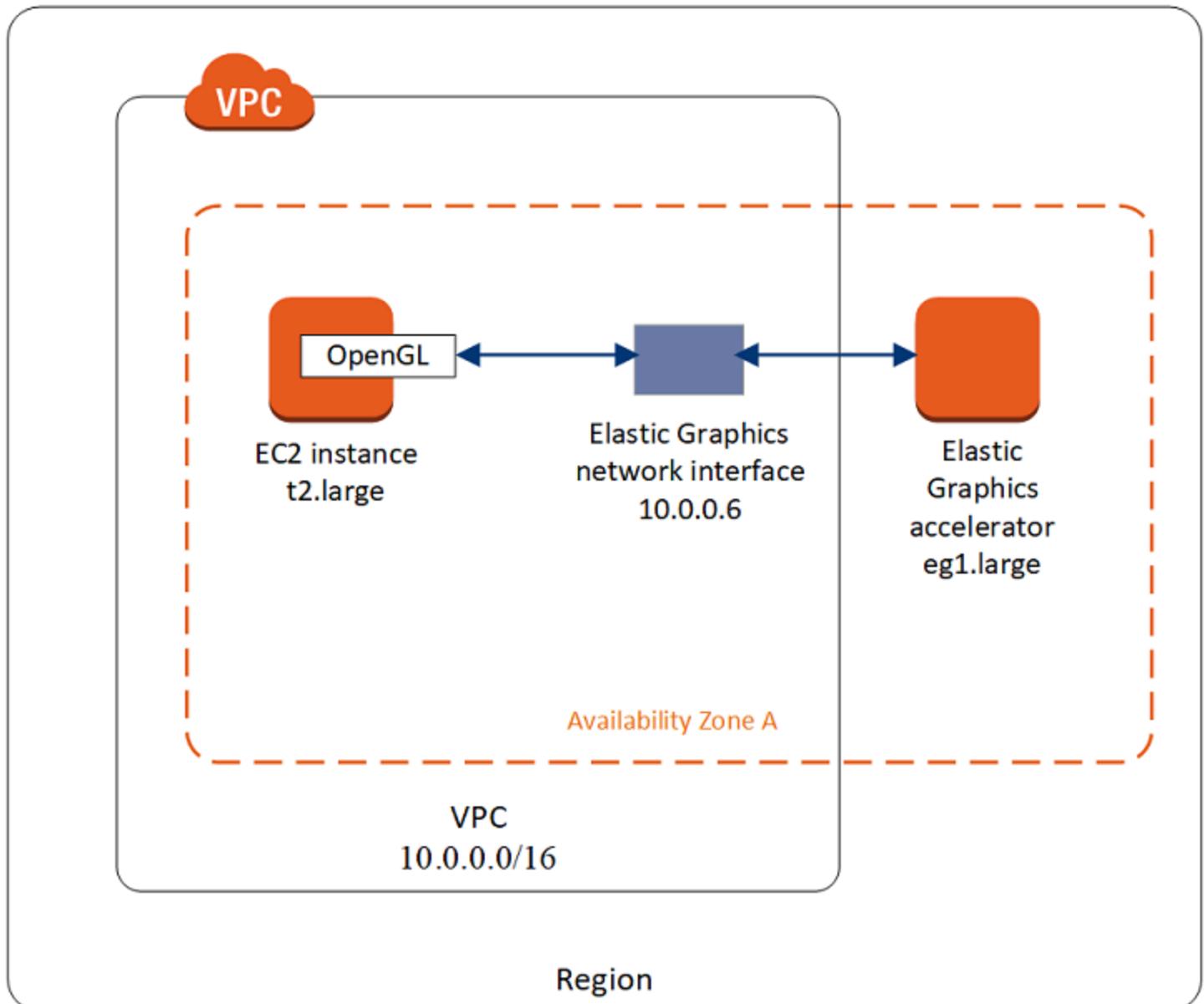
Akselerator Elastic Graphics berikut tersedia. Anda dapat melampirkan akselerator Elastic Graphics apa pun ke tipe instans yang didukung.

Akselerator Elastic Graphics	Memori grafis (GB)
<code>eg1.medium</code>	1
<code>eg1.large</code>	2
<code>eg1.xlarge</code>	4
<code>eg1.2xlarge</code>	8

Akselerator Elastic Graphics bukan bagian dari perangkat keras instans Anda. Sebaliknya, akselerator Elastic Graphics terhubung ke jaringan melalui antarmuka jaringan, yang dikenal sebagai

antarmuka jaringan Elastic Graphics. Saat Anda meluncurkan atau memulai ulang instans dengan akselerasi grafis, antarmuka jaringan Elastic Graphics akan dibuat di VPC untuk Anda.

Antarmuka jaringan Elastic Graphics dibuat di subnet dan VPC yang sama dengan instans Anda serta ditetapkan sebagai alamat privat IPv4 dari subnet itu. Akselerator yang dilampirkan ke instans Amazon EC2 Anda dialokasikan dari kumpulan akselerator yang tersedia di Zona Ketersediaan yang sama dengan instans Anda.



Akselerator Elastic Graphics mendukung standar API untuk API OpenGL 4.3 dan versi sebelumnya, yang dapat digunakan untuk aplikasi batch atau akselerasi grafis 3D. Pustaka OpenGL yang

dioptimalkan Amazon di instans Anda mendeteksi akselerator yang dilampirkan. Ini mengarahkan Panggilan API OpenGL dari instans Anda ke akselerator, yang kemudian memproses permintaan dan mengembalikan hasilnya. Lalu lintas antara instans dan akselerator menggunakan bandwidth yang sama dengan lalu lintas jaringan instans, jadi sebaiknya Anda memiliki bandwidth jaringan yang memadai. Konsultasikan vendor perangkat lunak Anda untuk pertanyaan kepatuhan dan versi OpenGL.

Secara default, grup keamanan default untuk VPC Anda dikaitkan dengan antarmuka jaringan Elastic Graphics. Lalu lintas jaringan Elastic Graphics menggunakan protokol TCP dan port 2007. Pastikan grup keamanan untuk instans Anda mengizinkan hal ini. Untuk informasi selengkapnya, lihat [Konfigurasi grup keamanan Anda](#).

Harga untuk Elastic Graphics

Anda dikenai biaya untuk setiap detik saat akselerator Elastic Graphics dilampirkan ke instans yang berstatus `running` saat akselerator berada dalam status `Ok`. Anda tidak dikenai biaya untuk akselerator yang dilampirkan ke instans yang berstatus `pending`, `stopping`, `stopped`, `shutting-down`, atau `terminated`. Anda juga tidak akan dikenai biaya saat akselerator berada dalam status `Unknown` atau `Impaired`.

Harga untuk akselerator hanya tersedia dengan tarif Sesuai Permintaan. Anda dapat melampirkan akselerator ke Instans Terpesan atau Instans Spot, tetapi, harga Sesuai Permintaan untuk akselerator berlaku.

Untuk informasi selengkapnya, lihat [Harga Amazon Elastic Graphics](#).

Batasan Elastic Graphics

Sebelum Anda mulai menggunakan akselerator Elastic Graphics, perhatikan batasan berikut:

- Anda dapat memasang akselerator hanya ke instans Windows dengan Microsoft Windows Server 2012 R2 atau versi yang lebih baru. Saat ini instans Linux tidak didukung.
- Anda dapat melampirkan satu akselerator ke satu instans dalam satu waktu.
- Anda dapat melampirkan akselerator hanya selama peluncuran instans. Anda tidak dapat melampirkan akselerator ke instans yang ada.
- Anda tidak dapat melakukan hibernasi pada instans dengan akselerator yang dilampirkan.
- Anda tidak dapat membagikan akselerator antara instans.
- Anda tidak dapat mencopot akselerator dari instans atau mentransfernya ke instans lain. Jika tidak lagi membutuhkan akselerator, Anda harus mengakhiri instans Anda. Untuk mengubah tipe

akselerator, buat AMI dari instans Anda, akhiri instans tersebut, dan luncurkan instans baru dengan spesifikasi akselerator yang berbeda.

- Satu-satunya versi yang didukung dari API OpenGL adalah versi 4.3 dan versi sebelumnya. DirectX, CUDA, dan OpenCL tidak didukung.
- Akselerator Elastic Graphics tidak terlihat atau dapat diakses melalui pengelola perangkat instans Anda.
- Anda tidak dapat memesan atau menjadwalkan kapasitas akselerator.

Bekerja dengan Elastic Graphics

Important

Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.

Anda dapat meluncurkan instans dan mengaitkannya dengan akselerator Elastic Graphics selama peluncuran. Kemudian, Anda harus menginstal pustaka yang diperlukan secara manual pada instans Anda yang memungkinkan komunikasi dengan akselerator. Untuk batasan, lihat [Batasan Elastic Graphics](#).

Tugas

- [Konfigurasi grup keamanan Anda](#)
- [Luncurkan instans dengan akselerator Elastic Graphics](#)
- [Instal perangkat lunak yang diperlukan untuk Elastic Graphics](#)
- [Verifikasi fungsionalitas Elastic Graphics pada instans Anda](#)
- [Lihat informasi Elastic Graphics](#)
- [Kirim umpan balik](#)

Konfigurasi grup keamanan Anda

Elastic Graphics membutuhkan grup keamanan yang memungkinkan lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri. Grup keamanan harus menyertakan aturan masuk dan keluar berikut ini.

Ke dalam

Tipe	Protokol	Port	Sumber
Elastic Graphics	TCP	2007	ID grup keamanan (ID sumber daya sendiri)

Ke luar

Tipe	Protokol	Rentang Port	Tujuan
Elastic Graphics	TCP	2007	ID grup keamanan (ID sumber daya sendiri)

Jika Anda menggunakan konsol Amazon EC2 untuk meluncurkan instans dengan akselerator Elastic Graphics, Anda dapat mengizinkan wizard peluncuran instans untuk membuat aturan grup keamanan yang diperlukan secara otomatis, atau Anda dapat memilih keamanan yang dibuat sebelumnya.

Jika Anda meluncurkan instans menggunakan AWS CLI atau SDK, Anda harus menentukan grup keamanan yang Anda buat sebelumnya.

Untuk membuat grup keamanan Elastic Graphics

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 `Elastic Graphics security group`.
 - b. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat grup keamanan.
 - c. Untuk VPC, pilih VPC yang ingin Anda gunakan Elastic Graphics.
 - d. Pilih Buat grup keamanan.
4. Di panel navigasi, pilih Grup Keamanan, pilih grup keamanan yang baru saja Anda buat, dan pada tab Detail, salin ID grup keamanan.
5. Pada tab Aturan ke dalam, pilih Edit aturan ke dalam, lalu lakukan hal-hal berikut ini:
 - a. Pilih Tambahkan aturan.

- b. Untuk Tipe, pilih Elastic Graphics.
 - c. Untuk Tipe sumber, pilih Kustom.
 - d. Untuk Sumber, tempelkan ID grup keamanan yang Anda salin sebelumnya.
 - e. Pilih Simpan aturan.
6. Pada tab Aturan ke luar, pilih Edit aturan ke luar, lalu lakukan hal-hal berikut ini:
- a. Pilih Tambahkan aturan.
 - b. Untuk Tipe, pilih Elastic Graphics.
 - c. Untuk Tipe tujuan, pilih Kustom.
 - d. Untuk Tujuan, tempelkan ID grup keamanan yang Anda salin sebelumnya.
 - e. Pilih Simpan aturan.

Untuk informasi selengkapnya, lihat [Grup keamanan Amazon EC2 untuk instans EC2 Anda](#).

Luncurkan instans dengan akselerator Elastic Graphics

Anda dapat mengaitkan akselerator Elastic Graphics ke instans selama peluncuran. Jika peluncuran gagal, kemungkinan alasannya adalah sebagai berikut:

- Kapasitas akselerator Elastic Graphics tidak memadai
- Melebihi batas akselerator Elastic Graphics di Wilayah
- Tidak punya alamat IPv4 privat yang cukup di VPC Anda untuk membuat antarmuka jaringan bagi akselerator

Untuk informasi selengkapnya, lihat [Batasan Elastic Graphics](#).

Untuk mengaitkan akselerator Elastic Graphics selama peluncuran instans (konsol)

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari dasbor, pilih Luncurkan instans.
3. Di bawah Nama dan tag, masukkan nilai untuk Nama. Anda dapat memilih Tambahkan tag tambahan secara opsional untuk menambahkan lebih banyak tag ke sumber daya yang terkait dengan instance yang Anda luncurkan.
4. Di bawah Gambar Aplikasi dan OS (Gambar Mesin Amazon), pilih AMI Windows.

5. Pada Tipe instans, pilih tipe instans yang didukung. Untuk informasi selengkapnya, lihat [Dasar-dasar Elastic Graphics](#).
6. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang sudah ada atau buat yang baru.
7. Di samping Pengaturan jaringan, pilih Edit, lalu tentukan pengaturan jaringan yang akan digunakan untuk instans Anda.
 - a. Untuk Jaringan, pilih VPC untuk instans Anda.
 - b. Untuk Subnet, pilih subnet untuk meluncurkan instance Anda.
 - c. Untuk Firewall (grup keamanan), Anda dapat menggunakan grup keamanan yang Anda buat secara manual [Konfigurasi grup keamanan Anda](#), atau membiarkan konsol membuat grup keamanan untuk Anda dengan aturan masuk dan keluar yang diperlukan. Tambahkan grup keamanan tambahan sesuai kebutuhan.
8. (Opsional) Di bawah Konfigurasi penyimpanan, konfigurasikan ukuran volume root Anda dan tambahkan volume tambahan sesuai kebutuhan.
9. Perluas bagian Detail lanjutan.
10. Di bawah Detail lanjutan, untuk GPU Elastis, pilih jenis akselerator Elastic Graphics.
11. Di panel Ringkasan, pilih Luncurkan instans.

Untuk mengaitkan akselerator Elastic Graphics selama peluncuran instans (AWS CLI)

Anda dapat menggunakan AWS CLI perintah [run-instance](#) dengan parameter berikut:

```
--elastic-gpu-specification Type=eg1.medium
```

Untuk parameter `--security-group-ids`, Anda harus menyertakan grup keamanan yang memiliki aturan masuk dan keluar yang diperlukan. Untuk informasi selengkapnya, lihat [Konfigurasi grup keamanan Anda](#).

Untuk mengaitkan akselerator Elastic Graphics selama peluncuran instans (Alat untuk Windows PowerShell)

Gunakan PowerShell perintah [New-EC2InstanceTools](#) untuk Windows.

Instal perangkat lunak yang diperlukan untuk Elastic Graphics

Jika Anda meluncurkan instans Anda menggunakan AMI AWS Windows saat ini, perangkat lunak yang diperlukan diinstal secara otomatis selama boot pertama. Jika Anda meluncurkan instans menggunakan AMI Windows yang tidak menginstal perangkat lunak yang diperlukan secara otomatis, Anda harus menginstal perangkat lunak yang diperlukan pada instans secara manual.

Untuk menginstal perangkat lunak yang diperlukan Elastic Graphics (jika perlu)

1. Hubungkan dengan instans.
2. Unduh [penginstal Elastic Graphics](#) dan buka. Manajer instalasi terhubung ke titik akhir Elastic Graphics dan mengunduh versi terbaru dari perangkat lunak yang diperlukan.

Note

Jika tautan unduhan tidak berfungsi, coba peramban lain, atau salin alamat tautan dan tempelkan ke tab peramban baru.

3. Lakukan boot ulang instans untuk memverifikasi.

Verifikasi fungsionalitas Elastic Graphics pada instans Anda

Paket Elastic Graphics pada instans Anda menyertakan alat yang dapat digunakan untuk melihat status akselerator, dan untuk memverifikasi bahwa perintah OpenGL dari instans Anda ke akselerator berfungsi.

Jika instans Anda diluncurkan dengan AMI yang belum menginstal paket Elastic Graphics, Anda dapat mengunduh dan menginstalnya sendiri. Untuk informasi selengkapnya, lihat [Instal perangkat lunak yang diperlukan untuk Elastic Graphics](#).

Anda dapat menggunakan salah satu metode berikut ini untuk memverifikasi fungsionalitas Elastic Graphics pada instans Anda.

Note

Jika monitor status Elastic Graphics atau alat baris perintah mengembalikan hasil yang tidak terduga, lihat [Menyelesaikan masalah status yang tidak sehat](#).

Elastic Graphics status monitor

Anda dapat menggunakan alat monitor status untuk melihat informasi tentang status akselerator Elastic Graphics yang dilampirkan. Secara default, alat ini tersedia di area notifikasi bilah tugas, di instans Windows Anda dan menunjukkan status akselerator grafis. Berikut ini adalah nilai-nilai yang memungkinkan.

Sehat

Akselerator Elastic Graphics aktif dan sehat.

Memperbarui

Status akselerator Elastic Graphics sedang diperbarui. Mungkin perlu beberapa menit untuk menampilkan status.

Keluar dari layanan

Akselerator Elastic Graphics keluar dari layanan. Untuk mendapatkan informasi tentang kesalahan selengkapnya, pilih [Baca Selengkapnya](#).

Elastic Graphics command line tool

Anda dapat menggunakan alat baris perintah Elastic Graphics, `egcli.exe`, untuk memeriksa status akselerator. Jika ada masalah dengan akselerator, alat akan mengembalikan pesan kesalahan.

Untuk meluncurkan alat tersebut, buka prompt perintah dari dalam instans Anda dan jalankan perintah berikut:

```
C:\Program Files\Amazon\EC2ElasticGPUs\manager\egcli.exe
```

Alat tersebut juga mendukung parameter berikut:

`--json, -j`

Menunjukkan apakah akan menampilkan pesan JSON atau tidak. Nilai yang mungkin adalah `true` dan `false`. Default-nya adalah `true`.

`--imds, -i`

Menunjukkan apakah akan memeriksa metadata instans untuk ketersediaan akselerator atau tidak. Nilai yang mungkin adalah `true` dan `false`. Default-nya adalah `true`.

Berikut ini adalah contoh output. Status dari OK menunjukkan bahwa akselerator aktif dan sehat.

```
EG Infrastructure is available.  
Instance ID egpu-f6d94dfa66df4883b284e96db7397ee6  
Instance Type eg1.large  
EG Version 1.0.0.885 (Manager) / 1.0.0.95 (OpenGL Library) / 1.0.0.69 (OpenGL  
  Redirector)  
EG Status: Healthy  
JSON Message:  
{  
  "version": "2016-11-30",  
  "status": "OK"  
}
```

Berikut ini adalah nilai-nilai yang mungkin untuk status:

OK

Akselerator Elastic Graphics aktif dan sehat.

UPDATING

Driver Elastic Graphics sedang diperbarui.

NEEDS_REBOOT

Driver Elastic Graphics telah diperbarui dan diperlukan boot ulang untuk instans Amazon EC2.

LOADING_DRIVER

Driver Elastic Graphics sedang dimuat.

CONNECTING_EGPU

Driver Elastic Graphics sedang memverifikasi konektivitas dengan akselerator Elastic Graphics.

ERROR_UPDATE_RETRY

Terjadi kesalahan saat memperbarui driver Elastic Graphics, pembaruan akan segera dicoba lagi.

ERROR_UPDATE

Terjadi kesalahan yang tidak dapat dipulihkan saat memperbarui driver Elastic Graphics.

ERROR_LOAD_DRIVER

Terjadi kesalahan saat memuat driver Elastic Graphics.

ERROR_EGPU_CONNECTIVITY

Akselerator Elastic Graphics tidak dapat dijangkau.

Lihat informasi Elastic Graphics

Anda dapat melihat informasi tentang akselerator Elastic Graphics yang dilampirkan ke instans Anda.

Untuk melihat informasi tentang akselerator Elastic Graphics (konsol)

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 Detail, temukan ID Elastic Graphics. Pilih ID untuk melihat informasi berikut tentang akselerator Elastic Graphics:
 - Status Lampiran
 - Tipe
 - Status kondisi

Untuk melihat informasi tentang akselerator Elastic Graphics (AWS CLI)

Anda dapat menggunakan [perintah describe-elastic-gpus](#) AWS CLI :

```
aws ec2 describe-elastic-gpus
```

Anda dapat menggunakan [AWS CLI perintah deskripsi-network-interface dan memfilter berdasarkan ID pemilik untuk melihat informasi tentang antarmuka](#) jaringan Elastic Graphics.

```
aws ec2 describe-network-interfaces --filters "Name=attachment.instance-owner-id,Values=amazon-elasticgpus"
```

Untuk melihat informasi tentang akselerator Elastic Graphics (Alat untuk Windows PowerShell)

Gunakan salah satu perintah berikut ini:

- [Get-EC2ElasticGpu](#)

- [Get-EC2NetworkInterface](#)

Untuk melihat informasi tentang akselerator Elastic Graphics menggunakan metadata instans

1. Hubungkan ke instans Windows Anda yang menggunakan akselerator Elastic Graphics.
2. Lakukan salah satu langkah berikut ini:
 - Dari PowerShell, gunakan cmdlet berikut:

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

- Dari peramban web Anda, tempel URL berikut ke dalam bidang alamat:

```
http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

Kirim umpan balik

Anda dapat mengirimkan umpan balik tentang pengalaman Anda dengan Elastic Graphics, sehingga tim dapat melakukan peningkatan lebih lanjut.

Untuk mengirimkan umpan balik menggunakan Monitor Status Elastic Graphics

1. Di area notifikasi bilah tugas, di instans Windows Anda, buka Status Monitor Elastic Graphics.
2. Di pojok kiri bawah, pilih Umpan Balik.
3. Masukkan umpan balik Anda dan pilih Kirim.

Pemeliharaan Elastic Graphics

Important

Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.

AWS mungkin menentukan bahwa akselerator Elastic Graphics dalam keadaan tidak sehat jika:

- Diperlukan pembaruan keamanan atau infrastruktur
- Diperlukan pembaruan perangkat lunak
- Ada masalah dengan host yang mendasarinya

Ketika AWS menentukan bahwa akselerator Elastic Graphics dalam keadaan tidak sehat, ia menjadwalkan akselerator untuk pensiun. AWS memberi tahu Anda tentang pensiun akselerator yang tertunda, dan memberi Anda langkah-langkah perbaikan yang perlu Anda ambil.

Topik

- [Bagaimana saya akan diberitahu?](#)
- [Apa yang harus saya lakukan?](#)
- [Apa yang terjadi ketika akselerator mencapai tanggal pensiunnya?](#)

Bagaimana saya akan diberitahu?

Saat AWS menjadwalkan akselerator Elastic Graphics untuk pensiun, ia mengirimkan pemberitahuan pensiun akselerator kepada Anda. [AWS Health Dashboard](#) AWS juga mengirim email ke alamat email yang terkait dengan AWS akun Anda. Ini adalah alamat email yang sama yang Anda gunakan untuk masuk ke AWS Management Console.

Note

Jika Anda menggunakan akun email yang tidak Anda periksa secara teratur, gunakan AWS Health Dashboard untuk menentukan apakah ada akselerator Elastic Graphics Anda yang dijadwalkan untuk pensiun. Anda juga dapat mengubah informasi kontak untuk AWS akun Anda di halaman [Pengaturan Akun](#).

Pemberitahuan pensiun mengatur hal-hal berikut ini:

- ID instans tempat akselerator dilampirkan
- Informasi tentang masalah yang berdampak pada akselerator
- Tanggal pensiun untuk akselerator
- Langkah-langkah perbaikan yang harus Anda lakukan

Apa yang harus saya lakukan?

Ketika diberi tahu bahwa akselerator Elastic Graphics Anda dijadwalkan untuk pensiun, Anda harus [berhenti dan memulai instans](#) yang akseleratornya dilampirkan agar akselerator yang lama dan tidak sehat diganti dengan akselerator yang baru dan sehat.

Sebaiknya Anda menutup aplikasi grafis yang berjalan pada instans sebelum Anda berhenti dan memulai ulang instans.

Important

Jika Anda tidak berhenti dan memulai instans sebelum tanggal pensiun yang dijadwalkan, akselerator yang terkait dengan instans Anda dihentikan secara otomatis, yang dapat menyebabkan aplikasi Anda berhenti berfungsi.

Anda harus berhenti dan memulai instans. Melakukan boot ulang instans tidak akan menggantikan akselerator yang tidak sehat dengan yang sehat.

Apa yang terjadi ketika akselerator mencapai tanggal pensiunnya?

Ketika akselerator Elastic Graphics yang tidak sehat mencapai tanggal pensiun yang dijadwalkan, AWS secara permanen menghentikannya. Untuk menerima pengganti akselerator Anda yang tidak sehat, baik sebelum maupun setelah tanggal pensiun, Anda harus berhenti dan memulai instans yang akseleratornya dilampirkan.

Jika Anda tidak berhenti dan memulai instans sebelum tanggal pensiun yang dijadwalkan, akselerator yang terkait dengan instans Anda dihentikan secara otomatis, yang dapat menyebabkan aplikasi Anda berhenti berfungsi.

Gunakan CloudWatch metrik untuk memantau Grafik Elastis

Important

Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.

Anda dapat memantau akselerator Elastic Graphics menggunakan Amazon CloudWatch, yang mengumpulkan metrik tentang kinerja akselerator Anda. Statistik ini dicatat dalam jangka waktu dua

minggu, sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang performa layanan Anda.

Secara default, akselerator Elastic Graphics mengirim data metrik ke CloudWatch dalam periode 5 menit.

Untuk informasi selengkapnya tentang Amazon CloudWatch, lihat [Panduan CloudWatch Pengguna Amazon](#).

Metrik Elastic Graphics

Namespace `AWS/ElasticGPUs` menyertakan metrik berikut untuk Elastic Graphics.

Metrik	Deskripsi
GPU Gagal ConnectivityCheck	Melaporkan apakah konektivitas ke akselerator Elastic Graphics aktif atau gagal. Nilai nol (0) menunjukkan bahwa koneksi aktif. Nilai satu (1) menunjukkan kegagalan konektivitas. Unit: Hitungan
GPU Gagal HealthCheck	Melaporkan apakah akselerator Elastic Graphics telah melewati pemeriksaan kondisi status dalam satu menit terakhir. Nilai nol (0) menunjukkan bahwa pemeriksaan status lulus. Nilai satu (1) menunjukkan kegagalan pemeriksaan status. Unit: Jumlah
GPU MemoryUtilization	Memori GPU yang digunakan. Unit: MiB

Dimensi Elastic Graphics

Anda dapat memfilter data metrik untuk akselerator Elastic Graphics menggunakan dimensi berikut.

Dimensi	Deskripsi
EGPUId	Memfilter data menurut akselerator Elastic Graphics.
InstanceId	Memfilter data menurut instans tempat akselerator Elastic Graphics dilampirkan.

Lihat CloudWatch metrik untuk Grafik Elastis

Metrik dikelompokkan berdasarkan namespace layanan terlebih dahulu, lalu dimensi yang didukung. Anda dapat menggunakan prosedur berikut untuk melihat metrik akselerator Elastic Graphics Anda.

Untuk melihat metrik Elastic Graphics menggunakan konsol CloudWatch

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Jika perlu, ubah Wilayah. Dari bilah navigasi, pilih Wilayah tempat akselerator Elastic Graphics Anda berada. Untuk informasi selengkapnya, lihat [Wilayah dan Titik Akhir](#).
3. Di panel navigasi, pilih Metrik.
4. Untuk Semua metrik, pilih Elastic Graphics, Metrik Elastic Graphics.

Untuk melihat metrik Elastic Graphics (AWS CLI)

Gunakan perintah [list-metrics](#) berikut:

```
aws cloudwatch list-metrics --namespace "AWS/ElasticGPUs"
```

Buat CloudWatch alarm untuk memantau Elastic Graphics

Anda dapat membuat CloudWatch alarm yang mengirimkan pesan Amazon SNS saat alarm berubah status. Alarm mengawasi satu metrik selama periode waktu yang Anda tentukan, dan mengirimkan notifikasi ke topik Amazon SNS berdasarkan nilai metrik relatif terhadap ambang batas tertentu selama beberapa periode waktu.

Misalnya, Anda dapat membuat alarm yang memantau kondisi akselerator Elastic Graphics dan mengirimkan notifikasi saat akselerator grafik gagal dalam pemeriksaan kondisi selama tiga periode 5 menit berturut-turut.

Untuk membuat alarm di status kondisi akselerator Elastic Graphics

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi, pilih Alarm, Buat Alarm.
3. Pilih Pilih metrik, Elastic Graphics, Metrik Elastic Graphics.
4. Pilih metrik HealthCheckGagal GPU dan pilih Select metric.
5. Konfigurasi alarm sebagai berikut:
 - a. Untuk Detail alarm, ketikkan nama dan deskripsi pada alarm Anda. Untuk Kapanpun, pilih \geq dan ketik 1.
 - b. Untuk Tindakan, pilih daftar notifikasi yang ada atau pilih Daftar baru.
 - c. Pilih Buat Alarm.

Pemecahan Masalah

Important

Amazon Elastic Graphics mencapai akhir masa pakai pada 8 Januari 2024. Untuk beban kerja yang memerlukan akselerasi grafis, sebaiknya gunakan instans Amazon EC2 G4ad, G4dn, atau G5.

Berikut ini adalah kesalahan umum dan langkah pemecahan masalah.

Daftar Isi

- [Menyelidiki masalah performa aplikasi](#)
 - [Masalah performa rendering OpenGL](#)
 - [Masalah performa akses jarak jauh](#)
- [Menyelesaikan masalah status yang tidak sehat](#)
 - [Periksa konfigurasi instans](#)
 - [Hentikan dan mulai instans](#)
 - [Verifikasi komponen yang diinstal](#)
 - [Periksa log Elastic Graphics](#)
- [Mengapa saya melihat banyak ENI?](#)

Menyelidiki masalah performa aplikasi

Elastic Graphics menggunakan jaringan instans untuk mengirim perintah OpenGL ke kartu grafis yang dilampirkan secara jarak jauh. Selain itu, desktop yang menjalankan aplikasi OpenGL dengan akselerator Elastic Graphics biasanya diakses menggunakan teknologi akses jarak jauh. Penting untuk membedakan antara masalah performa yang terkait dengan rendering OpenGL atau teknologi akses jarak jauh desktop.

Masalah performa rendering OpenGL

Performa rendering OpenGL ditentukan oleh jumlah perintah OpenGL dan bingkai yang dihasilkan pada instans jarak jauh.

Performa rendering dapat bervariasi bergantung pada faktor-faktor berikut:

- Performa akselerator Elastic Graphics
- Kinerja jaringan
- Performa CPU
- Model rendering, kompleksitas skenario
- Perilaku aplikasi OpenGL

Cara mudah untuk mengevaluasi performa adalah dengan menampilkan jumlah bingkai yang di-render pada instans jarak jauh. Akselerator Elastic Graphics menampilkan maksimum 25 FPS pada instans jarak jauh untuk mencapai kualitas terbaik sekaligus mengurangi penggunaan jaringan.

Untuk menunjukkan jumlah frame yang diproduksi

1. Buka file berikut di editor teks. Buat file jika file tidak ada.

```
C:\Program Files\Amazon\EC2ElasticGPUs\conf\eg.conf
```

2. Identifikasi bagian [Application], atau tambahkan jika tidak ada, dan tambahkan parameter konfigurasi berikut:

```
[Application]  
show_fps=1
```

3. Mulai ulang aplikasi dan periksa lagi FPS.

Jika FPS mencapai 15-25 FPS saat memperbarui adegan yang di-render, akselerator Elastic Graphics bekerja dengan maksimal. Masalah performa lain yang Anda alami kemungkinan besar terkait dengan akses jarak jauh ke desktop instans. Jika demikian, lihat bagian Masalah Performa Akses Jarak Jauh.

Jika nomor FPS lebih rendah dari 15, Anda dapat mencoba yang berikut ini:

- Tingkatkan performa akselerator Elastic Graphics dengan memilih tipe akselerator grafik yang lebih bertenaga.
- Tingkatkan performa jaringan secara menyeluruh menggunakan kiat-kiat berikut:
 - Periksa jumlah bandwidth yang masuk dan keluar ke dan dari titik akhir akselerator Elastic Graphics. Titik akhir akselerator Elastic Graphics dapat diambil dengan perintah berikut:
PowerShell

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/meta-data/elastic-gpus/associations/[ELASTICGPU_ID]).content
```

- Lalu lintas jaringan dari instans ke titik akhir akselerator Elastic Graphics berkaitan dengan volume perintah yang dihasilkan aplikasi OpenGL.
- Lalu lintas jaringan dari titik akhir akselerator Elastic Graphics ke instans berkaitan dengan jumlah bingkai yang dihasilkan oleh akselerator grafis.
- Jika Anda melihat penggunaan jaringan mencapai throughput jaringan maksimum instans, coba gunakan instans dengan jatah throughput jaringan yang lebih tinggi.
- Tingkatkan performa CPU:
 - Aplikasi mungkin memerlukan banyak sumber daya CPU selain yang dibutuhkan oleh akselerator Elastic Graphics. Jika Windows Task Manager melaporkan penggunaan sumber daya CPU yang tinggi, coba gunakan instans dengan lebih banyak daya CPU.

Masalah performa akses jarak jauh

Instans dengan akselerator Elastic Graphics yang dilampirkan dapat diakses menggunakan teknologi akses jarak jauh yang berbeda. Performa dan kualitas dapat bervariasi bergantung pada:

- Teknologi akses jarak jauh
- Performa instans
- Performa klien
- Latensi dan bandwidth jaringan antara klien dan instans

Pilihan yang memungkinkan untuk protokol akses jarak jauh meliputi:

- Remote Desktop Connection Microsoft
- NICE DCV
- VNC

Untuk informasi tentang optimisasi, lihat protokol khusus.

Menyelesaikan masalah status yang tidak sehat

Jika akselerator Elastic Graphics berstatus tidak sehat, gunakan langkah-langkah pemecahan masalah berikut ini untuk menyelesaikan masalah tersebut.

Periksa konfigurasi instans

Jika alat baris perintah Elastic Graphics, `egcli.exe`, mengembalikan output yang serupa dengan yang berikut ini, pastikan bahwa [grup keamanan Anda dikonfigurasi dengan benar](#) dan Anda meluncurkan instans dengan Layanan Metadata Instans yang aktif.

```
EG Version 1.0.7.4240 (Manager) / N/A (OpenGL Library) / N/A (OpenGL Redirector)
EG Status: Out Of Service
Something prevented the EG Infrastructure to work properly.
```

Hentikan dan mulai instans

Jika akselerator Elastic Graphics Anda dalam status tidak sehat, menghentikan instans dan memulainya lagi adalah opsi yang paling sederhana. Untuk informasi selengkapnya, lihat [Hentikan dan mulai instans Anda secara manual](#).

Warning

Ketika Anda menghentikan suatu instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan instans, pastikan untuk mencadangkannya ke penyimpanan persisten.

Verifikasi komponen yang diinstal

Buka Windows Control Panel dan konfirmasi bahwa komponen berikut ini diinstal:

- Amazon Elastic Graphics Manager
- Pustaka OpenGL Amazon Elastic Graphics
- Pengalih OpenGL GPU Amazon EC2 Elastic

Jika salah satu item ini hilang, Anda harus menginstalnya secara manual. Untuk informasi selengkapnya, lihat [Instal perangkat lunak yang diperlukan untuk Elastic Graphics](#).

Periksa log Elastic Graphics

Buka Windows Event Viewer, perbesar bagian Log Aplikasi dan Layanan, lalu cari kesalahan di log peristiwa berikut ini:

- EC2ElasticGPUs
- EC2ElasticGPUs GUI

Mengapa saya melihat banyak ENI?

Saat memanggil [StartInstances](#) instans EC2 dengan akselerator Elastic Graphics, Elastic Network Interface (ENI) baru dibuat pada instance untuk memungkinkan perintah OpenGL dikirim ke kartu grafis yang terpasang dari jarak jauh.

Jika Anda menelepon [StartInstances](#) berkali-kali dalam waktu singkat (beberapa detik atau kurang) pada instans EC2 yang sama, antarmuka jaringan baru dibuat pada setiap panggilan. Namun:

- Hanya satu antarmuka jaringan yang akan digunakan oleh akselerator Elastic Graphics.
- Antarmuka jaringan tambahan tidak dikenai biaya apa pun dan akan dirilis secara otomatis dalam 24 jam.

Menginstal WSL pada instans Windows Anda

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.

Untuk informasi selengkapnya tentang WSL, lihat [Dokumentasi Subsistem Windows untuk Linux](#) di situs web Microsoft Build.

Batasan

- WSL tersedia dalam dua versi: WSL 1 dan WSL 2.
 - Untuk instans `.meta1` EC2, Anda dapat menginstal WSL 1 atau WSL 2.
 - Untuk instans EC2 tervirtualisasi, Anda harus menginstal WSL 1.
- Untuk sistem operasi Windows Server, WSL hanya dapat diinstal pada instans yang menjalankan berikut ini:
 - Windows Server 2019
 - Windows Server 2022

Pasang WSL

Instruksi berikut menginstal WSL pada instans EC2 yang menjalankan Windows Server 2022. Untuk petunjuk menginstal WSL pada instans EC2 yang menjalankan Windows Server 2019, lihat [Menginstal WSL pada versi Windows Server sebelumnya](#) di situs web Microsoft. Setelah mengikuti petunjuk tersebut, Anda dapat menggunakan langkah 3 dalam petunjuk di bawah ini untuk mengonfigurasi WSL untuk menggunakan WSL 1.

Instal WSL 1

1. Untuk menginstal WSL, jalankan perintah instalasi standar berikut pada instans EC2 Anda, tetapi pastikan untuk mengaktifkan WSL 1 dengan menyertakan `--enable-wsl1`. Secara default, WSL 2 diinstal. Jika instans Anda diluncurkan menggunakan tipe instans tervirtualisasi, Anda harus menyelesaikan langkah 3 dalam prosedur ini untuk menyetel versi ke WSL 1.

```
ws1 --install --enable-wsl1 --no-launch
```

2. Mulai ulang instans EC2 Anda.

```
shutdown -r -t 20
```

3. Untuk mengonfigurasi WSL untuk menggunakan WSL 1, jalankan perintah berikut pada instans Anda. Untuk informasi selengkapnya tentang mengatur versi WSL, lihat [Langkah instalasi manual untuk WSL versi sebelumnya](#) di situs web Microsoft Build.

```
wsl --set-default-version 1
```

4. Instal distribusi default.

```
wsl --install
```

Instal WSL 2

- Untuk menginstal WSL, jalankan perintah instalasi standar berikut di instans EC2 Anda. Secara default, WSL 2 diinstal. Jika Anda menginstal WSL pada sebuah `.metal` instans, maka ini adalah satu-satunya langkah untuk melakukan.

```
wsl --install
```

Untuk informasi selengkapnya, lihat [Menginstal Linux di Windows dengan WSL](#) di situs web Microsoft Build.

Mutakhirkan instans Amazon EC2 Windows ke versi Windows Server yang lebih baru

Ada dua metode untuk memutakhirkan versi Windows Server sebelumnya yang berjalan pada sebuah instance: peningkatan dan migrasi di tempat (juga disebut side-by-side pemutakhiran). Pemutakhiran langsung akan memutakhirkan file sistem operasi sementara pengaturan dan file pribadi Anda akan tetap utuh. Migrasi melibatkan pengambilan pengaturan, konfigurasi, dan data serta melakukan porting hal tersebut ke sistem operasi yang lebih baru pada instans Amazon EC2 baru.

Microsoft secara tradisional merekomendasikan migrasi ke versi Windows Server yang lebih baru daripada memutakhirkan. Migrasi dapat menghasilkan lebih sedikit kesalahan atau masalah pemutakhiran, tetapi bisa memakan waktu lebih lama daripada pemutakhiran langsung karena kebutuhan untuk menyediakan instans baru, merencanakan dan mem-port aplikasi, serta menyesuaikan pengaturan konfigurasi pada instans baru. Pemutakhiran langsung bisa lebih cepat, tetapi ketidaksesuaian perangkat lunak dapat menghasilkan kesalahan.

Daftar Isi

- [Lakukan pemutakhiran di tempat pada instans Windows Anda](#)
- [Lakukan pemutakhiran otomatis pada instans Windows Anda](#)
- [Migrasikan instance Windows ke jenis instans generasi saat ini](#)
- [Asisten replatforming Windows ke Linux untuk database Microsoft SQL Server](#)
- [Memecahkan masalah pemutakhiran pada instance Windows](#)

Lakukan pemutakhiran di tempat pada instans Windows Anda

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 AWS PV, Intel Network Adapter, atau driver 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.
 - [Opsinya Pemutakhiran untuk Windows Server 2012](#)
 - [Opsinya Pemutakhiran untuk Windows Server 2012 R2](#)
 - [Opsinya pemutakhiran dan konversi untuk Windows Server 2016](#)
 - [Opsinya pemutakhiran dan konversi untuk Windows Server 2019](#)
 - [Opsinya peningkatan dan konversi untuk Windows Server 2022](#)
 - [Pusat Pemutakhiran 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 [Ubah tipe instans](#).
- 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 [Volume Elastis Amazon EBS](#) 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. Tipe perangkat lunak ini dapat bertentangan dengan proses pemutakhiran. Aktifkan kembali perangkat lunak antivirus dan anti-spyware serta firewall setelah pemutakhiran versi selesai.
- Perbarui ke driver terbaru seperti yang dijelaskan di [Migrasikan instance Windows ke jenis instans generasi saat ini](#).
- 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 Enhanced Networking

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 AMI yang didukung Amazon EBS-Backed](#).
2. Pastikan instans Windows Server Anda menggunakan driver jaringan terbaru.
 - a. Untuk memperbarui driver AWS PV Anda, lihat [Mutakhirkan driver PV pada instans Windows](#).
 - b. Untuk memperbarui driver ENA Anda, lihat [Instal driver Elastic Network Adapter \(ENA\)](#).

- c. Untuk memperbarui driver Intel, lihat [Aktifkan jaringan yang disempurnakan dengan antarmuka Intel 82599 VF pada instans EC2](#)
3. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
4. Di panel navigasi, pilih Contoh. Temukan instans tersebut. Catat ID instans dan Zona Ketersediaan untuk instans tersebut. Anda membutuhkan informasi ini nanti dalam prosedur ini.
5. Jika Anda meningkatkan dari Windows Server 2012 atau 2012 R2 ke Windows Server 2016, 2019, atau 2022, lakukan hal berikut pada instans Anda sebelum melanjutkan:
 - a. Hapus instalasi layanan EC2Config. Untuk informasi selengkapnya, lihat [Hentikan, mulai ulang, hapus, atau uninstal EC2Config](#).
 - b. Instal EC2Launch v1 atau agen EC2Launch v2. Untuk informasi selengkapnya, lihat [Konfigurasi instans Windows menggunakan EC2Launch](#) dan [Konfigurasi instans Windows menggunakan EC2Launch v2](#).
 - c. Instal Agen AWS Systems Manager SSM. Untuk informasi selengkapnya, lihat [Bekerja dengan SSM Agent](#) dalam Panduan Pengguna AWS Systems Manager .
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 samping 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 spanduk Berhasil membuat volume vol-**1234567890example**, 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 [Buat 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 SKU penginstalan lengkap untuk instans Windows Server Anda, dan pilih Berikutnya.
 - 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 akan gagal pada 1 dari 2 pemeriksaan status selama proses pemutakhiran. Saat pemutakhiran selesai, kedua pemeriksaan

status lolos. Anda dapat memeriksa log sistem untuk keluaran konsol atau menggunakan CloudWatch metrik Amazon 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 kedua pemeriksaan status setelah beberapa jam, lihat [Memecahkan masalah pemutakhiran pada instance 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 [Konfigurasi instans Windows menggunakan EC2Launch](#) dan [Konfigurasi instans Windows menggunakan EC2Launch v2](#).
3. Jika Anda memutakhirkan ke Windows Server 2012 R2, kami sarankan Anda meningkatkan driver PV ke driver AWS PV. Jika Anda memutakhirkan pada instans berbasis Nitro, kami menyarankan Anda untuk menginstal atau memutakhirkan driver NVME dan ENA. Untuk informasi selengkapnya, lihat [Windows Server 2012 R2](#), [Instal atau tingkatkan driver AWS NVMe menggunakan PowerShell](#), atau [Mengaktifkan jaringan yang ditingkatkan di Windows](#).
4. Aktifkan kembali perangkat lunak antivirus dan anti-spyware serta firewall.

Lakukan pemutakhiran otomatis pada instans Windows Anda

Anda dapat melakukan pemutakhiran otomatis instance Windows dan SQL Server Anda AWS dengan runbook AWS Systems Manager Automation.

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 sumber daya Anda AWS secara terpusat. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna [AWS Systems Manager](#).
- **AWS Systems Manager Agen (Agen SSM)** 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 Runbook SSM.** Runbook SSM menentukan tindakan yang dilakukan Systems Manager pada instans yang Anda kelola. Runbook SSM menggunakan JavaScript Object Notation (JSON) atau YANG, 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 pengaturan Multiakun dan Wilayah, menentukan bagaimana otomatisasi Anda bercabang.

- Target

Pilih target yang ingin Anda terapkan otomatisasi. Pengaturan ini juga diatur dalam pengaturan Multiakun dan Wilayah.

- Nilai Parameter

Gunakan nilai yang ditentukan dalam parameter dokumen otomatisasi.

- Grup Sumber Daya

Di AWS, sumber daya adalah entitas yang dapat Anda gunakan. Contohnya termasuk instans Amazon EC2, AWS CloudFormation tumpukan, atau bucket Amazon S3. Jika Anda bekerja dengan banyak 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 pengaturan Multiakun dan Wilayah. Saat Anda menetapkan parameter kontrol tarif, Anda menentukan berapa banyak armada Anda yang akan menerapkan otomatisasi, baik berdasarkan jumlah target maupun persentase armada.

Multiakun dan Wilayah

Selain parameter yang ditentukan dalam Kontrol Tarif yang juga digunakan dalam pengaturan Multiakun 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 di mana 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 upgrade 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 AWS Identity and Access Management Panduan Pengguna.
- [Pilih opsi bagaimana Anda ingin otomatisasi dijalankan](#). Opsi untuk eksekusi adalah Eksekusi simpel, Kontrol nilai, Multiakun dan Wilayah, serta Eksekusi 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 instans 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 ke Layanan AWS seperti Amazon S3 dan akses untuk mengunduh tambalan 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 instans 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. Klik kanan ISO dan pasang ke instans sebagai, 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 otomatisasi [AWSEC2- CloneInstance AndUpgrade Windows](#).

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 Systems Manager terhadap instans AWS Amazon EC2 dan AMI. Untuk informasi selengkapnya, lihat [Buat Profil Instans IAM untuk Systems Manager](#) dalam Panduan Pengguna AWS Systems Manager .

- `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 tambalan).

- `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 melewati 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 peningkatan 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 simpel, Kontrol nilai, Multiakun dan Wilayah, serta Eksekusi 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. Klik kanan ISO dan pasang ke instans sebagai, 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 runbook [AWSEC2- CloneInstance AndUpgrade SqlServer](#) untuk melakukan pemutakhiran 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 meningkatkan SQL Server Anda menggunakan runbook otomatisasi [AWSEC2- CloneInstance AndUpgrade SQLServer](#).

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. Hal ini memungkinkan Systems Manager untuk berkomunikasi dengan instans 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. Lihat [Buat Profil Instans IAM untuk Systems Manager](#) di Panduan Pengguna AWS Systems Manager .
5. Di AWS Systems Manager konsol, di panel navigasi kiri, pilih Instans Terkelola. 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 tambalan).

- `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 diatur ke `true`, Systems Manager melakukan boot ulang pada instans sebelum membuat AMI untuk peningkatan.

- `TargetSQLVersion`

Tipe: String

(Opsional) Versi SQL Server target. Default-nya adalah `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.

Migrasikan instance Windows ke jenis instans generasi saat ini

AMI AWS Windows dikonfigurasi dengan pengaturan default yang digunakan oleh media instalasi Microsoft, dengan beberapa penyesuaian. Kustomisasi mencakup driver dan konfigurasi yang mendukung jenis instans generasi terbaru, yang merupakan [instance yang dibangun pada Sistem AWS Nitro](#), seperti M5 atau C5.

Saat bermigrasi ke instans berbasis Nitro, termasuk instans bare metal, kami menyarankan Anda mengikuti 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

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 2008 R2 dan setelahnya. Untuk memigrasikan instance Linux ke jenis instans generasi terbaru, lihat [the section called “Ubah tipe instans”](#)

Daftar Isi

- [Bagian 1: Instal dan tingkatkan driver AWS PV](#)
- [Bagian 2: Instal dan mutakhirkan ENA](#)
- [Bagian 3: Tingkatkan driver AWS NVMe](#)
- [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 driver AWS PV, ENA, dan NVMe menggunakan AWS Systems Manager](#)
- [Migrasikan instance Windows dari jenis instans Nitro ke Xen](#)

Note

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 driver AWS PV, ENA, dan NVMe menggunakan AWS Systems Manager](#), lalu lanjutkan dengan Bagian 4 dan Bagian 5.

Sebelum Anda memulai

Prosedur ini mengasumsikan bahwa Anda sedang menjalankan jenis instans berbasis Xen generasi sebelumnya, seperti M4 atau C4, dan Anda bermigrasi ke [instance](#) yang dibangun di Sistem Nitro. AWS

Anda harus menggunakan PowerShell versi 3.0 atau yang lebih baru untuk berhasil melakukan upgrade.

Note

Saat bermigrasi ke instans generasi terbaru, IP statis atau pengaturan jaringan DNS kustom pada ENI yang ada mungkin hilang karena instans tersebut akan menggunakan perangkat Adaptor Jaringan yang Ditingkatkan secara default.

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 ke generasi terbaru. Namun, kami menyarankan Anda menyelesaikannya sebelum bermigrasi jika Anda bermigrasi secara khusus ke jenis instans bare metal.

Bagian 1: Instal dan tingkatkan driver AWS PV

Meskipun driver AWS PV tidak digunakan dalam sistem Nitro, Anda masih harus memutakhirkannya jika Anda menggunakan versi sebelumnya dari Citrix PV atau PV. AWS Driver AWS PV terbaru menyelesaikan masalah bug di versi driver sebelumnya yang mungkin muncul saat Anda menggunakan sistem Nitro, atau jika Anda perlu bermigrasi kembali ke instans berbasis Xen. Sebagai praktik terbaik, kami sarankan untuk selalu memperbarui ke driver terbaru untuk instance Windows. AWS

Gunakan prosedur berikut untuk melakukan peningkatan driver AWS PV di tempat, atau untuk meningkatkan 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 [Mutakhirkan driver PV pada instans Windows](#).

Untuk memutakhirkan Kontroler Domain, lihat [Tingkatkan pengontrol domain \(peningkatan AWS PV\)](#).

Untuk melakukan upgrade atau ke driver AWS 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 Elastic Network Adapter \(ENA\) pada instans EC2](#).

1. [Unduh](#) driver terbaru ke instans.
2. Ekstrak arsip zip.
3. Instal driver dengan menjalankan `install.ps1` PowerShell skrip dari folder yang diekstraksi.

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 Elastic Network Adapter \(ENA\) pada instans EC2](#).

Bagian 3: Tingkatkan driver AWS NVMe

AWS Driver NVMe digunakan untuk berinteraksi dengan Amazon EBS dan volume penyimpanan instans SSD yang diekspos sebagai perangkat blok NVMe dalam sistem Nitro untuk kinerja yang lebih baik.

Important

Instruksi berikut dimodifikasi secara khusus saat Anda menginstal atau memutakhirkan AWS NVMe pada instance generasi sebelumnya dengan maksud untuk memigrasikan instance ke jenis instans generasi terbaru.

1. [Unduh](#) paket driver terbaru ke instans.
2. Ekstrak arsip zip.
3. Instal driver dengan menjalankan `dpinst.exe`.
4. Buka PowerShell sesi 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 (x86) versi 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 instans, ubah tipe instans menjadi instans generasi terbaru dan mulai, lalu lanjutkan ke Bagian 4. Jika Anda memulai lagi instans pada tipe instans generasi sebelumnya sebelum bermigrasi ke tipe instans generasi terbaru, itu tidak akan bisa 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 instans.
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 [Konfigurasi instans Windows menggunakan EC2Launch](#).

Bagian 5: Instal driver port serial untuk instans bare metal

Tipe instans `i3.metal` menggunakan perangkat serial berbasis PCI daripada perangkat serial berbasis port I/O. AMI Windows terbaru secara otomatis menggunakan perangkat serial berbasis PCI dan menginstal driver port serial. Jika Anda tidak menggunakan instans yang diluncurkan dari AMI Windows yang disediakan Amazon tertanggal 2018.04.11 atau 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 prompt perintah 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 instans yang diluncurkan dari AMI Windows yang disediakan Amazon tertanggal 2018.11.19 atau setelahnya, maka Anda harus menginstal driver menggunakan Intel Chipset INF Utility.

Untuk menginstal driver chipset

1. [Unduh utilitas chipset](#) ke instans.
2. Ekstrak file.
3. Jalankan `SetupChipset.exe`.
4. Terima perjanjian lisensi perangkat lunak Intel dan instal driver chipset.
5. Boot ulang instans.

(Alternatif) Tingkatkan driver AWS PV, ENA, dan NVMe menggunakan AWS Systems Manager

Dokumen otomatisasi `AWSSupport-UpgradeWindowsAWSDrivers` mengotomatiskan langkah-langkah yang dijelaskan di Bagian 1, Bagian 2, dan Bagian 3. Metode ini juga dapat memperbaiki instans di mana pemutakhiran driver gagal.

Dokumen `AWSSupport-UpgradeWindowsAWSDrivers` otomatisasi meningkatkan atau memperbaiki penyimpanan dan AWS driver jaringan pada instans EC2 yang ditentukan. Dokumen ini mencoba menginstal versi terbaru AWS driver online dengan menghubungi 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 \(peningkatan AWS PV\)](#).

Untuk secara otomatis meningkatkan driver AWS PV, ENA, dan NVMe menggunakan AWS Systems Manager

1. Buka konsol Systems Manager di <https://console.aws.amazon.com/systems-manager>.
2. Pilih Otomasi, Eksekusi Otomasi.
3. Cari dan kemudian pilih dokumen `AWSSupport- UpgradeWindows AWSDrivers` otomatisasi, lalu pilih Jalankan otomatisasi.
4. Di bagian Parameter Input, konfigurasi 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 instans, 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.

Migrasikan instance Windows dari jenis instans Nitro ke Xen

Prosedur berikut mengasumsikan bahwa Anda sedang menjalankan jenis instans berbasis Nitro, dan Anda bermigrasi ke instance berdasarkan Sistem Xen, seperti M4 atau C4. Misalnya spesifikasi jenis, lihat Panduan [Jenis Instans Amazon EC2](#). Lakukan langkah-langkah berikut sebelum melakukan migrasi untuk menghindari kesalahan selama proses booting.

Untuk bermigrasi dari Nitro ke Xen

1. Cadangkan data Anda.
2. Verifikasi bahwa [kebijakan Windows san](#) Anda memungkinkan volume penyimpanan non-root untuk online.
3. AWS Driver PV harus diinstal dan ditingkatkan pada instans Nitro sebelum Anda bermigrasi ke instans Xen. Untuk langkah-langkah menginstal dan meningkatkan driver AWS PV, lihat [Bagian 1: Instal dan tingkatkan driver AWS PV](#).
4. Perbarui ke EC2Launch v2 versi terbaru. Untuk langkah-langkahnya, lihat [Migrasikan ke EC2Launch v2](#).
5. Buka PowerShell sesi dan jalankan perintah berikut sebagai administrator untuk sysprep driver perangkat. Menjalankan sysprep memastikan bahwa driver penyimpanan boot awal yang diperlukan untuk melakukan boot pada instans Xen terdaftar dengan benar dengan Windows.

Note

Menjalankan perintah menggunakan versi PowerShell (x86) akan menghasilkan kesalahan. Perintah ini menambahkan hanya driver perangkat yang penting untuk boot pada basis data perangkat yang penting. Itu tidak menjalankan persiapan sysprep lengkap.

```
Start-Process rundll32.exe sppnp.dll, Sysprep_Generalize_Pnp -wait
```

6. Lakukan migrasi ke tipe instans Xen saat proses sysprep selesai.

Asisten replatforming Windows ke Linux untuk database Microsoft SQL Server

Untuk informasi tentang replatforming database Microsoft SQL Server dari Windows ke Linux, lihat [asisten replatforming Windows ke Linux untuk Database Microsoft SQL Server di Panduan Pengguna Microsoft SQL Server](#) di Amazon EC2.

Memecahkan masalah pemutakhiran pada instance Windows

AWS menyediakan dukungan upgrade untuk masalah atau masalah dengan Upgrade Helper Service, sebuah AWS utilitas yang membantu Anda melakukan upgrade 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 instans tidak lulus kedua pemeriksaan status setelah beberapa jam, periksa hal berikut.

- Jika Anda meningkatkan ke Windows Server 2008 dan kedua pemeriksaan status gagal setelah beberapa jam, peningkatan mungkin telah gagal dan menampilkan prompt Klik OK untuk mengonfirmasi pembatalan. Karena konsol tidak dapat diakses pada status ini, tombol tersebut tidak dapat diklik. Untuk menyiasatinya, lakukan boot ulang 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 instans tidak lulus 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 terkait pemutakhiran atau migrasi sistem operasi, sebaiknya tinjau artikel yang tercantum di [Sebelum Anda memulai pemutakhiran langsung](#).

Armada EC2 dan Armada Spot

Armada EC2 dan Armada Spot didesain untuk menjadi cara yang berguna untuk meluncurkan armada—atau grup—instans dengan AWS. Setiap instance dalam armada didasarkan pada [template peluncuran](#) atau serangkaian parameter peluncuran yang Anda konfigurasi secara manual saat peluncuran.

Armada menyediakan fitur dan keuntungan berikut. Keuntungan ini memungkinkan Anda memaksimalkan penghematan biaya serta mengoptimalkan ketersediaan dan performa saat menjalankan aplikasi pada banyak instans EC2.

Berbagai tipe instans dan opsi pembelian

Dalam satu panggilan API, armada dapat meluncurkan banyak tipe instans dan opsi pembelian (Instans Spot dan Sesuai Permintaan), sehingga memungkinkan Anda mengoptimalkan biaya melalui penggunaan Instans Spot. Anda juga dapat memanfaatkan diskon Instans Terpesan dan Savings Plans dengan menggunakannya bersama dengan Instans Sesuai Permintaan di armada.

Mendistribusikan instans di seluruh Zona Ketersediaan

Armada secara otomatis mencoba mendistribusikan instans secara merata di banyak Zona Ketersediaan untuk ketersediaan tinggi. Hal ini memberikan ketahanan jika Zona Ketersediaan menjadi tidak tersedia.

Penggantian otomatis Instans Spot

Armada yang menyertakan Instans Spot dapat secara otomatis meminta kapasitas Spot pengganti jika Instans Spot Anda terinterupsi atau mengalami gangguan karena perubahan kondisi instans. Melalui Penyeimbangan Ulang Kapasitas, armada juga dapat memantau dan secara proaktif mengganti Instans Spot Anda yang memiliki risiko interupsi tinggi.

Armada EC2 adalah pilihan yang baik jika Anda memerlukan fleksibilitas untuk mengelola aspek siklus hidup instans atau mekanisme penskalaan. Anda juga dapat menggunakan Armada Spot, tetapi kami tidak menyarankan Anda melakukannya karena ini adalah API warisan tanpa investasi yang direncanakan. Namun, jika sudah menggunakan Armada Spot, Anda dapat terus menggunakannya. Armada Spot dan Armada EC2 menawarkan fungsionalitas inti yang sama.

i Tip

Sebagai praktik terbaik secara umum, sebaiknya luncurkan armada Instans Spot dan Sesuai Permintaan dengan Auto Scaling Amazon EC2 karena menyediakan fitur tambahan yang dapat Anda gunakan untuk mengelola armada Anda. Daftar fitur tambahan mencakup penggantian pemeriksaan kondisi otomatis untuk Instans Spot dan Sesuai Permintaan, pemeriksaan kondisi berbasis aplikasi, dan integrasi dengan Elastic Load Balancing untuk memastikan distribusi lalu lintas aplikasi yang merata ke instans berkondisi baik milik Anda. Anda juga dapat menggunakan grup Auto Scaling saat menggunakan AWS layanan seperti Amazon ECS, Amazon EKS (grup node yang dikelola sendiri), dan Amazon VPC Lattice. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon EC2 Auto Scaling](#).

Topik

- [Armada EC2](#)
- [Armada Spot](#)
- [Pantau peristiwa armada menggunakan Amazon EventBridge](#)
- [Tutorial untuk Armada EC2 dan Armada Spot](#)
- [Contoh konfigurasi Armada EC2 dan Armada Spot](#)
- [Kuota armada](#)

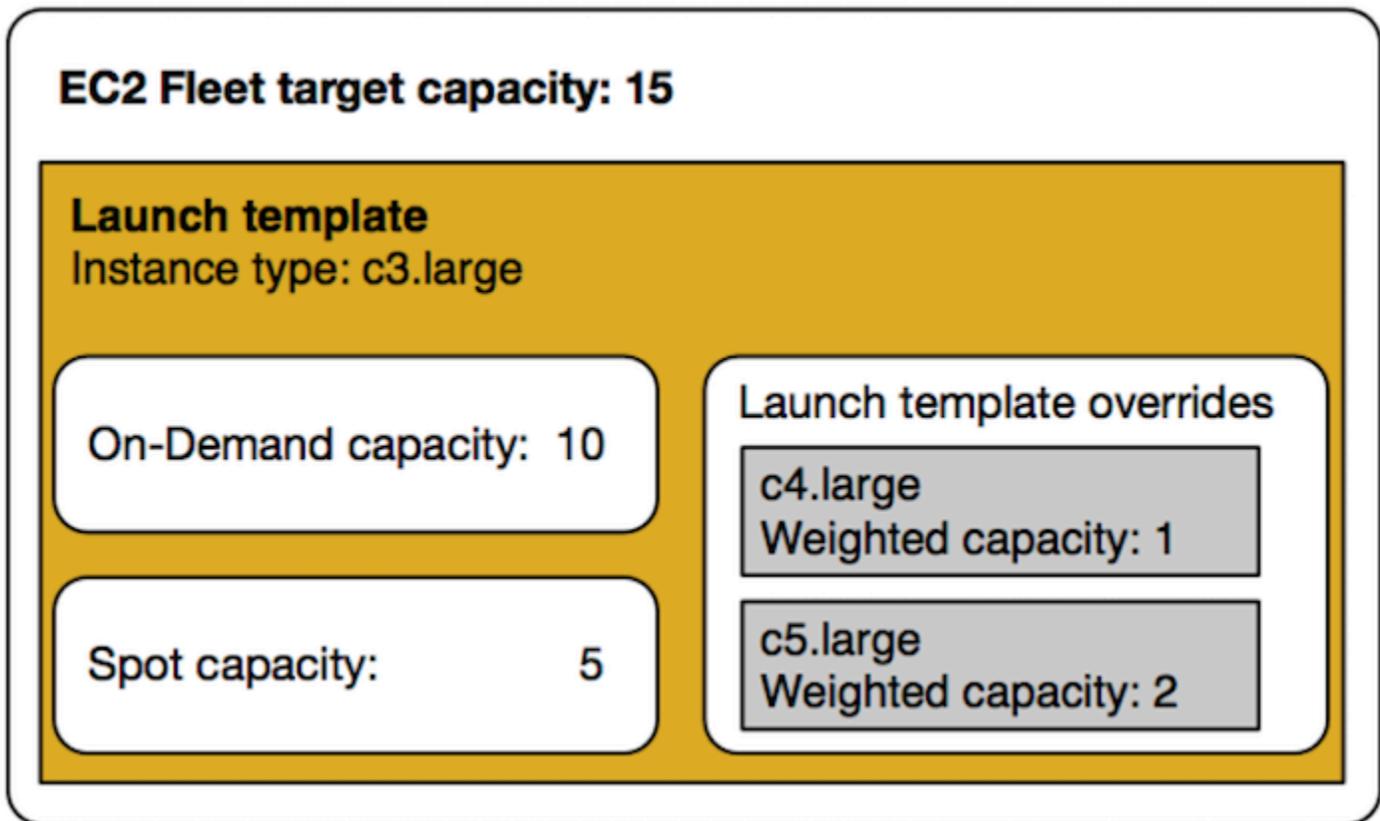
Armada EC2

Armada EC2 berisi informasi konfigurasi untuk meluncurkan armada instans. Dalam satu panggilan API, armada dapat meluncurkan banyak tipe instans di banyak Zona Ketersediaan dengan menggunakan opsi pembelian Instans Spot, Instans Sesuai Permintaan, Instans Terpesan, dan Savings Plans secara bersama-sama. Dengan Armada EC2, Anda dapat:

- Menentukan target kapasitas Spot dan Sesuai Permintaan terpisah serta jumlah maksimum yang tersedia Anda bayarkan per jam
- Menentukan tipe instans yang paling sesuai untuk aplikasi Anda
- Menentukan cara Amazon EC2 harus mendistribusikan kapasitas armada Anda dalam setiap opsi pembelian

Anda juga dapat mengatur jumlah maksimum per jam yang bersedia Anda bayarkan untuk armada Anda, dan Armada EC2 meluncurkan instans hingga mencapai jumlah maksimum. Saat jumlah maksimum yang ingin Anda bayarkan tercapai, armada akan berhenti meluncurkan instans meskipun belum memenuhi kapasitas target.

Armada EC2 berupaya meluncurkan jumlah instans yang diperlukan untuk memenuhi kapasitas target yang ditentukan dalam permintaan Anda. Jika Anda menentukan total harga maksimum per jam, kapasitas akan dipenuhi hingga mencapai jumlah maksimum yang bersedia Anda bayarkan. Armada tersebut juga dapat berupaya mempertahankan kapasitas Spot targetnya jika Instans Spot Anda terinterupsi. Untuk informasi selengkapnya, lihat [Cara kerja Instans Spot](#).



Anda dapat menentukan tipe instans dalam jumlah tidak terbatas per Armada EC2. Tipe instans tersebut dapat disediakan menggunakan opsi pembelian Spot dan Sesuai Permintaan. Anda juga dapat menentukan lebih dari satu Zona Ketersediaan, menentukan harga Spot maksimum yang berbeda untuk setiap instans, dan memilih opsi Spot tambahan untuk setiap armada. Amazon EC2 menggunakan opsi tertentu untuk menyediakan kapasitas saat armada diluncurkan.

Saat armada berjalan, jika Amazon EC2 mengeklaim kembali Instans Spot karena kenaikan harga atau kegagalan instans, Armada EC2 dapat mencoba untuk mengganti instans dengan tipe instans apa pun yang Anda tentukan. Hal ini memudahkan untuk mendapatkan kembali kapasitas selama

lonjakan harga Spot. Anda dapat mengembangkan strategi sumber daya yang fleksibel dan elastis untuk setiap armada. Misalnya, dalam armada tertentu, kapasitas primer Anda dapat bersifat Sesuai Permintaan yang dilengkapi dengan kapasitas Spot yang lebih terjangkau jika tersedia.

Jika Anda memiliki Instans Terpesan dan Anda menentukan Instans Sesuai Permintaan dalam armada Anda, Armada EC2 akan menggunakan Instans Terpesan Anda. Misalnya, jika armada Anda menentukan Instans Sesuai Permintaan sebagai `c4.large`, dan Anda memiliki Instans Terpesan untuk `c4.large`, Anda akan menerima harga Instans Terpesan. Hal yang sama berlaku jika Anda menggunakan Savings Plans.

Tidak ada biaya tambahan untuk penggunaan Armada EC2. Anda hanya membayar untuk instans EC2 yang diluncurkan armada untuk Anda.

Daftar Isi

- [Batasan Armada EC2](#)
- [Instans performa yang dapat melonjak](#)
- [Tipe permintaan Armada EC2](#)
- [Strategi konfigurasi Armada EC2](#)
- [Bekerja dengan Armada EC2](#)

Batasan Armada EC2

Batasan berikut berlaku untuk Armada EC2:

- Armada EC2 hanya tersedia melalui [API Amazon EC2](#), [AWS CLI](#), [AWS SDK](#), dan [AWS CloudFormation](#).
- Permintaan Armada EC2 tidak dapat menjangkau AWS Wilayah. Anda perlu membuat Armada EC2 terpisah untuk setiap Wilayah.
- Permintaan Armada EC2 tidak dapat menjangkau subnet yang berbeda dari Zona Ketersediaan yang sama.

Instans performa 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 mengakumulasi 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.

Tipe permintaan Armada EC2

Terdapat tiga tipe permintaan Armada EC2:

`instant`

Jika Anda mengonfigurasi tipe permintaan sebagai `instant`, Armada EC2 akan mengajukan permintaan satu kali sinkron untuk kapasitas yang Anda inginkan. Dalam responsnya, API mengembalikan instans yang diluncurkan, bersama dengan kesalahan untuk instans yang tidak dapat diluncurkan. Untuk informasi selengkapnya, lihat [Gunakan Armada EC2 tipe 'instan'](#).

`request`

Jika Anda mengonfigurasi tipe permintaan sebagai `request`, Armada EC2 akan mengajukan permintaan satu kali asinkron untuk kapasitas yang Anda inginkan. Setelah itu, jika kapasitas berkurang karena interupsi Spot, armada tidak akan berupaya untuk mengisi Instans Spot, juga tidak akan mengirimkan permintaan dalam kolam kapasitas Spot alternatif jika kapasitas tidak tersedia.

maintain

(Default) Jika Anda mengonfigurasi tipe permintaan sebagai `maintain`, Armada EC2 akan mengajukan permintaan asinkron untuk kapasitas yang Anda inginkan, dan mempertahankan kapasitas dengan secara otomatis mengisi ulang setiap Instans Spot yang terinterupsi.

Ketiga tipe permintaan mendapatkan keuntungan dari strategi alokasi. Untuk informasi selengkapnya, lihat [Strategi alokasi untuk Instans Spot](#).

Gunakan Armada EC2 tipe 'instan'

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 antrian.

Anda dapat menggunakan Armada EC2 tipe instan untuk meluncurkan instans ke Blok Kapasitas. Untuk informasi selengkapnya, lihat [Tutorial: 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 jenis [CreateFleet](#) permintaan sebagai `instant`. Untuk informasi selengkapnya, lihat [Membuat Armada EC2](#). Perhatikan bahwa setelah melakukan panggilan API, Anda tidak dapat memodifikasinya.

2. Jika Anda melakukan panggilan API, Armada EC2 akan mengajukan permintaan satu kali sinkron untuk kapasitas yang Anda inginkan.
3. Respons API mencantumkan instans yang diluncurkan, bersama dengan kesalahan untuk instans yang tidak dapat diluncurkan.
4. Anda dapat mendeskripsikan Armada EC2, mencantumkan instans yang terkait dengan Armada EC2, dan melihat riwayat Armada EC2.
5. Setelah instans diluncurkan, Anda dapat [menghapus permintaan armada](#). Saat menghapus permintaan armada, Anda juga dapat memilih untuk mengakhiri instans terkait, atau membiarkannya berjalan.
6. Anda dapat mengakhiri instans kapan saja.

Contoh

Contoh berikut menunjukkan cara menggunakan Armada EC2 tipe instant pada berbagai kasus penggunaan. Untuk informasi selengkapnya tentang penggunaan parameter CreateFleet API EC2, lihat [CreateFleet](#) di 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: Meluncurkan Instans Spot dalam satu Zona Ketersediaan](#)
- [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 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 penyimpanan 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 penyimpanan templat peluncuran. Templat peluncuran diidentifikasi dengan nama templat dan nomor versi peluncurannya. 12 penyimpanan 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 kolom kapasitas Spot berdasarkan strategi alokasi yang dioptimalkan kapasitas, untuk meluncurkan Instans Spot dari kolom 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 tipe instans yang berbeda dan bobot yang berbeda berdasarkan berapa banyak unit beban kerja yang dapat dijalankan pada instans dengan asumsi unit beban kerja memerlukan memori 15 GB dan 4 vCPU. Misalnya m5.xlarge (4 vCPU dan 16 GB memori) dapat menjalankan satu unit dan berbobot 1, m5.2xlarge (8 vCPU 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 [Pembobotan instans Armada EC2](#).

```
{
  "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: Meluncurkan Instans Spot dalam satu Zona Ketersediaan

Anda dapat mengonfigurasi armada untuk meluncurkan semua instance dalam satu Availability Zone dengan menyetel opsi Spot SingleAvailabilityZone ke true.

12 penyimpanan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing-masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangannya 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 penempatan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing-masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangannya 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 kolom 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.

12 penempatan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing-masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangannya sama. Total kapasitas target dan kapasitas target minimum keduanya diatur ke 20 instans, opsi pembelian default adalah spot, strategi alokasi Spot dioptimalkan kapasitas. Armada EC2 meluncurkan 20 Instans Spot dari kolam kapasitas Spot dengan kapasitas optimal menggunakan penempatan templat peluncuran, hanya jika dapat meluncurkan semua 20 instans pada saat yang bersamaan.

```
{
  "SpotOptions": {
    "AllocationStrategy": "capacity-optimized",
    "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 menempa templat peluncuran, memiliki tipe instans dan subnet yang berbeda (masing-masing di Zona Ketersediaan yang terpisah), tetapi kapasitas tertimbangannya 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-9x1",
    "Version": "$Latest"
  },
  "Overrides": [
    {
      "InstanceType": "c5n.9xlarge",
      "SubnetId": "subnet-fae8c380",
      "WeightedCapacity": 36
    },
    {
      "InstanceType": "c5n.9xlarge",
      "SubnetId": "subnet-e7188bab",
      "WeightedCapacity": 36
    },
    {
      "InstanceType": "c5n.9xlarge",
      "SubnetId": "subnet-49e41922",
      "WeightedCapacity": 36
    }
  ]
},
{
  "LaunchTemplateSpecification": {
    "LaunchTemplateName": "ec2-fleet-lt-4x1",
    "Version": "$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 $\text{TotalTargetCapacity} - \text{OnDemandTargetCapacity} = \text{DefaultTargetCapacityType}$, 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.

Armada tersebut memiliki 6 Reservasi Kapasitas yang tidak terpakai dalam 2 kolam. Jumlah Reservasi Kapasitas di setiap kolam ditunjukkan oleh `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-1t1",
        "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 Instans Sesuai Permintaan m5.large di us-east-1a – m5.large in us-east-1a diprioritaskan kedua, dan terdapat 3 Reservasi Kapasitas c3.large yang tidak terpakai.
- 10 Instans Spot dari salah satu dari 12 kolom kapasitas Spot yang memiliki kapasitas optimal sesuai dengan strategi alokasi yang dioptimalkan kapasitas.

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 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 penimpanan 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"
}
```

Strategi konfigurasi Armada EC2

Armada EC2 adalah grup Instans Sesuai Permintaan dan Instans Spot. Armada EC2 juga dapat berupa grup instans Blok Kapasitas.

Instans Sesuai Permintaan dan Instans Spot

Armada EC2 mencoba meluncurkan jumlah instans yang diperlukan untuk memenuhi kapasitas target yang ditentukan dalam permintaan Anda. Armada hanya dapat terdiri atas Instans Sesuai Permintaan, hanya Instans Spot, atau kombinasi dari Instans Sesuai Permintaan dan Instans Spot. Permintaan Instans Spot terpenuhi jika terdapat kapasitas yang tersedia dan harga maksimum per jam untuk permintaan Anda melebihi harga Spot. Armada tersebut juga berupaya mempertahankan kapasitas targetnya jika Instans Spot Anda terinterupsi.

Anda juga dapat mengatur jumlah maksimum per jam yang bersedia Anda bayarkan untuk armada Anda, dan Armada EC2 meluncurkan instans hingga mencapai jumlah maksimum. Saat jumlah maksimum yang ingin Anda bayarkan tercapai, armada akan berhenti meluncurkan instans meskipun belum memenuhi kapasitas target.

Kolam kapasitas spot adalah satu set instans EC2 yang tidak digunakan dengan tipe instans dan Zona Ketersediaan yang sama. Saat membuat Armada EC2, Anda dapat menyertakan beberapa spesifikasi peluncuran, yang bervariasi menurut tipe instans, Zona Ketersediaan, subnet, dan harga maksimum. Armada memilih kolam Instans Spot yang digunakan untuk memenuhi permintaan, berdasarkan spesifikasi peluncuran yang disertakan dalam permintaan Anda, dan konfigurasi permintaannya. Instans Spot berasal dari kolam yang dipilih.

Armada EC2 memungkinkan Anda menyediakan kapasitas EC2 dalam jumlah besar yang masuk akal untuk aplikasi Anda berdasarkan jumlah core atau instans, atau jumlah memori. Misalnya, Anda dapat menentukan Armada EC2 untuk meluncurkan kapasitas target 200 instans, yang 130 di antaranya adalah Instans Sesuai Permintaan dan sisanya adalah Instans Spot.

Instans Blok Kapasitas

Blok Kapasitas untuk ML memungkinkan Anda untuk memesan instans GPU di masa mendatang untuk mendukung beban kerja machine learning (ML) berdurasi pendek. Instans yang berjalan di Blok Kapasitas secara otomatis ditempatkan berdekatan di dalam [Amazon UltraClusters EC2](#). Untuk informasi selengkapnya tentang Blok Kapasitas, lihat [Blok Kapasitas untuk ML](#).

Gunakan strategi konfigurasi yang sesuai untuk membuat Armada EC2 yang memenuhi kebutuhan Anda.

Daftar Isi

- [Merencanakan Armada EC2](#)
- [Strategi alokasi untuk Instans Spot](#)
- [Pemilihan tipe instans berbasis atribut untuk Armada EC2](#)
- [Mengonfigurasi Armada EC2 untuk pencadangan Sesuai Permintaan](#)
- [Penyeimbangan Ulang Kapasitas](#)
- [Penimpaan harga maksimum](#)
- [Kontrol pengeluaran](#)
- [Pembobotan instans Armada EC2](#)

Merencanakan Armada EC2

Saat merencanakan Armada EC2, kami menyarankan Anda untuk melakukan hal berikut:

- Tentukan apakah Anda ingin membuat Armada EC2 yang mengirimkan permintaan satu kali sinkron atau asinkron untuk kapasitas target yang diinginkan, atau yang mempertahankan kapasitas target dari waktu ke waktu. Untuk informasi selengkapnya, lihat [Tipe permintaan Armada EC2](#).
- Tentukan tipe instans yang memenuhi kebutuhan aplikasi Anda.
- Jika Anda berencana untuk menyertakan Instans Spot di Armada EC2, tinjau [Praktik Terbaik Spot](#) sebelum Anda membuat armada. Gunakan praktik terbaik ini saat Anda merencanakan armada sehingga Anda dapat menyediakan instans dengan harga serendah mungkin.
- Menentukan kapasitas target Armada EC2 Anda. Anda dapat menetapkan kapasitas target dalam instans atau dalam unit kustom. Untuk informasi selengkapnya, lihat [Pembobotan instans Armada EC2](#).

- Tentukan pembagian dari kapasitas target Armada EC2 yang harus berupa kapasitas Sesuai Permintaan dan kapasitas Spot. Anda dapat menentukan 0 untuk kapasitas Sesuai Permintaan atau kapasitas Spot, atau keduanya.
- Tentukan harga Anda per unit, jika Anda menggunakan pembobotan instans. Untuk menghitung harga per unit, bagi harga per jam instans dengan jumlah unit (atau bobot) yang diwakili oleh instans ini. Jika Anda tidak menggunakan pembobotan instans, harga default per unit adalah harga per jam instans.
- Tentukan jumlah maksimum per jam yang ingin Anda bayarkan untuk armada. Untuk informasi selengkapnya, lihat [Kontrol pengeluaran](#).
- Tinjau opsi yang memungkinkan untuk Armada EC2 Anda. Untuk informasi tentang parameter armada, lihat [create-fleet](#) di Referensi Perintah AWS CLI . Untuk contoh konfigurasi Armada EC2, lihat [Contoh konfigurasi Armada EC2](#).

Strategi alokasi untuk Instans Spot

Konfigurasi peluncuran Anda menentukan semua kemungkinan kolam kapasitas Spot (tipe instans dan Zona Ketersediaan) tempat Armada EC2 dapat meluncurkan Instans Spot. Namun, saat meluncurkan instans, Armada EC2 menggunakan strategi alokasi yang Anda tentukan untuk memilih kolam tertentu dari semua kemungkinan kolam Anda.

Note

(Hanya instans Linux) Jika Anda mengonfigurasi Instans Spot untuk diluncurkan dengan [AMD SEV-SNP](#) diaktifkan, Anda akan mengganti biaya penggunaan per jam tambahan yang setara dengan 10% dari tarif [per jam On-Demand untuk jenis instans yang dipilih](#). Jika strategi alokasi menggunakan harga sebagai input, Armada EC2 tidak termasuk biaya tambahan ini; hanya harga Spot yang digunakan.

Strategi alokasi

Anda dapat menentukan salah satu strategi alokasi berikut untuk Instans Spot:

`price-capacity-optimized`(direkomendasikan)

Armada EC2 mengidentifikasi kolam dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami

yakini memiliki peluang interupsi terendah dalam waktu dekat. Armada EC2 kemudian meminta Instans Spot dari harga terendah dari kolam ini.

Strategi alokasi `price-capacity-optimized` adalah pilihan terbaik untuk sebagian besar beban kerja Spot, seperti aplikasi terkontainerisasi tanpa status, layanan mikro, aplikasi web, pekerjaan data dan analitik, serta pemrosesan batch.

`capacity-optimized`

Armada EC2 mengidentifikasi kolam dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami yakini memiliki peluang interupsi terendah dalam waktu dekat. Anda dapat secara opsional menetapkan prioritas untuk setiap tipe instans dalam armada menggunakan `capacity-optimized-prioritized`. Armada EC2 mengoptimalkan kapasitas terlebih dahulu, tetapi mempertimbangkan prioritas tipe instans dengan upaya terbaik.

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 `capacity-optimized` secara otomatis meluncurkan Instans Spot ke dalam kolam yang paling tersedia dengan melihat data kapasitas waktu nyata dan memprediksi kolam 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 `capacity-optimized` dapat menurunkan biaya keseluruhan beban kerja Anda.

Atau, Anda dapat menggunakan strategi alokasi `capacity-optimized-prioritized` dengan parameter prioritas untuk mengurutkan tipe instans dari prioritas tertinggi ke terendah. Anda dapat mengatur prioritas yang sama untuk tipe instans yang berbeda. Armada EC2 akan mengoptimalkan kapasitas terlebih dahulu, tetapi akan mempertimbangkan prioritas tipe instans 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. Perhatikan bahwa ketika Anda menetapkan prioritas untuk `capacity-optimized-prioritized`, prioritas yang sama akan diterapkan pada Instans Sesuai Permintaan jika `AllocationStrategy` Sesuai Permintaan diatur menjadi `prioritized`.

diversified

Instans Spot didistribusikan di semua kolam kapasitas Spot.

lowest-price(tidak disarankan)

Warning

Kami tidak merekomendasikan strategi `lowest-price` alokasi karena memiliki risiko interupsi tertinggi untuk Instans Spot Anda.

Instans Spot berasal dari kolam dengan harga terendah yang memiliki kapasitas tersedia. Ini adalah strategi default. Namun, kami menyarankan Anda mengganti default dengan menentukan strategi alokasi `price-capacity-optimized`.

Jika kolam dengan harga terendah tidak memiliki kapasitas yang tersedia, Instans Spot akan berasal dari kolam dengan harga terendah berikutnya yang memiliki kapasitas tersedia.

Jika kolam kehabisan kapasitas sebelum memenuhi kapasitas yang Anda inginkan, Armada EC2 akan terus memenuhi permintaan Anda dengan menarik dari kolam dengan 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.

InstancePoolsToUseCount

Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Berlaku hanya jika strategi alokasi diatur ke `lowest-price`. Armada EC2 memilih kolam Spot dengan harga terendah dan mengalokasikan kapasitas Spot target Anda secara merata di seluruh kolam Spot yang Anda tentukan.

Perhatikan bahwa Armada EC2 mencoba untuk menarik Instans Spot dari sejumlah kolam yang Anda tentukan dengan upaya terbaik. Jika kolam kehabisan kapasitas Spot sebelum memenuhi kapasitas yang Anda inginkan, Armada EC2 akan terus memenuhi permintaan Anda dengan menarik dari kolam dengan 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 kolom yang Anda tentukan.

Memilih strategi alokasi yang tepat

Anda dapat mengoptimalkan armada untuk kasus penggunaan dengan memilih strategi alokasi Spot yang sesuai. Untuk kapasitas target Instans Sesuai Permintaan, Armada EC2 selalu memilih tipe instans yang paling murah berdasarkan harga Sesuai Permintaan publik, sambil mengikuti strategi alokasi—baik `price-capacity-optimized`, `capacity-optimized`, `diversified`, ataupun `lowest-price`—untuk Instans Spot.

Menyeimbangkan harga terendah dan ketersediaan kapasitas

Untuk menyeimbangkan kompromi antara kolam kapasitas Spot dengan harga terendah dan kolam kapasitas Spot dengan ketersediaan kapasitas tertinggi, sebaiknya gunakan strategi alokasi `price-capacity-optimized`. Strategi ini membuat keputusan terkait kolam yang akan meminta Instans Spot dari berdasarkan harga kolam dan ketersediaan kapasitas Instans Spot di kolam tersebut. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami yakini memiliki kemungkinan interupsi paling rendah dalam waktu dekat, dengan tetap mempertimbangkan harga.

Jika armada Anda menjalankan beban kerja yang tangguh dan tanpa status, termasuk aplikasi terkontainerisasi, layanan mikro, aplikasi web, pekerjaan data dan analitik, serta pemrosesan batch, maka gunakan strategi alokasi `price-capacity-optimized` untuk penghematan biaya yang optimal dan ketersediaan kapasitas.

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 operasi titik pemeriksaan, Anda membuat strategi alokasi `price-capacity-optimized` cocok untuk beban kerja karena strategi ini mengalokasikan kapasitas dari kolam dengan harga terendah yang juga menawarkan tingkat interupsi Instans Spot yang rendah.

Untuk contoh konfigurasi yang menggunakan strategi alokasi `price-capacity-optimized`, lihat [Contoh 10: Luncurkan Instans Spot di armada `price-capacity-optimized`](#) .

Ketika beban kerja memiliki biaya interupsi yang tinggi

Anda dapat menggunakan strategi `capacity-optimized` secara opsional jika menjalankan beban kerja yang menggunakan tipe instans dengan harga yang sama, atau jika biaya interupsi

sangat signifikan sehingga penghematan biaya apa pun tidak memadai jika dibandingkan dengan peningkatan marginal dalam interupsi. 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. Untuk contoh konfigurasi yang menggunakan strategi alokasi `capacity-optimized`, lihat [Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas](#).

Ketika kemungkinan interupsi harus diminimalkan tetapi preferensi untuk tipe instans tertentu menjadi penting, Anda dapat mengekspresikan prioritas kolam Anda dengan menggunakan strategi alokasi `capacity-optimized-prioritized`, lalu mengatur urutan tipe instans yang akan digunakan dari prioritas tertinggi ke terendah. Untuk contoh konfigurasi, lihat [Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas](#).

Perhatikan bahwa ketika Anda menetapkan prioritas untuk `capacity-optimized-prioritized`, maka prioritas yang sama juga akan diterapkan pada Instans Sesuai Permintaan Anda jika `AllocationStrategy On-Demand` diatur ke `prioritized`.

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 `price-capacity-optimized` untuk memaksimalkan penghematan biaya sekaligus tetap mempertimbangkan ketersediaan kapasitas.

Jika armada Anda besar atau berjalan untuk waktu yang lama

Jika armada Anda berjumlah besar atau berjalan untuk waktu yang lama, Anda dapat meningkatkan ketersediaan armada dengan mendistribusikan Instans Spot di banyak kolam menggunakan strategi `diversified`. Misalnya, jika Armada EC2 Anda menentukan 10 kolam dan target kapasitas 100 instans, armada tersebut akan meluncurkan 10 Instans Spot di setiap kolam. 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 `diversified`, Armada EC2 tidak meluncurkan Instans Spot ke dalam kolam mana pun dengan harga Spot yang sama atau lebih tinggi dari [harga Sesuai Permintaan](#).

Mempertahankan kapasitas target

Setelah Instans Spot diakhiri karena perubahan harga Spot atau kapasitas yang tersedia dari kolam kapasitas Spot, Armada EC2 tipe `maintain` meluncurkan Instans Spot pengganti. Strategi alokasi menentukan kolam tempat instans pengganti diluncurkan, sebagai berikut:

- Jika strategi alokasinya adalah `price-capacity-optimized`, armada akan meluncurkan instans pengganti di kolam yang memiliki ketersediaan kapasitas Instans Spot paling banyak sekaligus juga mempertimbangkan harga dan mengidentifikasi kolam dengan harga terendah dengan ketersediaan kapasitas yang tinggi.
- Jika strategi alokasinya adalah `capacity-optimized`, armada akan meluncurkan instans pengganti di kolam yang memiliki ketersediaan kapasitas Instans Spot terbanyak.
- Jika strategi alokasinya adalah `diversified`, armada akan mendistribusikan Instans Spot pengganti di seluruh kolam yang tersisa.

Pemilihan tipe instans berbasis atribut untuk Armada EC2

Ketika membuat Armada EC2, Anda harus menentukan satu atau lebih tipe instans untuk mengonfigurasi Instans Sesuai Permintaan dan Instans Spot di 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 vCPU minimum dan maksimum yang diperlukan untuk instans Anda, dan Armada EC2 akan meluncurkan instans menggunakan tipe instans yang tersedia yang memenuhi persyaratan vCPU tersebut.

Pemilihan tipe instans berbasis atribut sangat ideal untuk beban kerja dan kerangka kerja yang fleksibel dalam menentukan tipe instans yang digunakan, seperti ketika menjalankan kontainer atau armada web, memproses big data, dan mengimplementasikan alat integrasi dan deployment berkelanjutan (CI/CD).

Keuntungan

Pemilihan tipe instans berbasis atribut memiliki keuntungan berikut:

- Mudah menggunakan jenis instans yang tepat — 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 EC2, 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 penyimpanan templat peluncuran.

- Penggunaan otomatis tipe instans baru — Saat Anda menentukan atribut instance daripada tipe instans, armada Anda dapat menggunakan tipe instance generasi yang lebih baru saat dirilis, “pemeriksaan masa depan” konfigurasi armada.
- Fleksibilitas tipe instans — Saat Anda menentukan atribut instans daripada tipe instans, Armada EC2 dapat memilih dari berbagai jenis instans untuk meluncurkan Instans Spot, yang mengikuti [praktik terbaik Spot dari fleksibilitas tipe instans](#).

Topik

- [Cara kerja pemilihan tipe instans berbasis atribut](#)
- [Perlindungan harga](#)
- [Pertimbangan](#)
- [Membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut](#)
- [Contoh konfigurasi 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. Armada EC2 akan meluncurkan instans pada tipe instans yang tersedia yang memiliki atribut instans yang ditentukan.

Topik

- [Tipe atribut instans](#)
- [Tempat mengonfigurasi pemilihan tipe instans berbasis atribut](#)
- [Cara Armada EC2 menggunakan pemilihan tipe 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 instance.
- Memori — Minimum dan GiBs maksimum memori per instance.

- Penyimpanan lokal — Apakah akan menggunakan EBS atau volume penyimpanan instans untuk penyimpanan lokal.
- Kinerja burstable — Apakah akan menggunakan keluarga instans T, termasuk tipe T4G, T3a, T3, dan T2.

Untuk deskripsi setiap atribut dan nilai default, lihat [InstanceRequirements](#) di Referensi API Amazon EC2.

Tempat mengonfigurasi pemilihan tipe instans berbasis atribut

Bergantung pada apakah Anda menggunakan konsol atau konsol AWS CLI, Anda dapat menentukan atribut instance untuk pemilihan jenis instans berbasis atribut sebagai berikut:

Di konsol, Anda dapat menentukan atribut instans dalam komponen konfigurasi armada berikut:

- Dalam templat peluncuran, lalu referensikan templat peluncuran dalam permintaan armada

Di dalam AWS CLI, Anda dapat menentukan atribut instance dalam 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, tipe instans yang berbeda dapat menggunakan prosesor berbasis x86 dan Arm.

Cara Armada EC2 menggunakan pemilihan tipe instans berbasis atribut saat menyediakan armada

Armada EC2 menyediakan armada dengan cara berikut:

- Armada EC2 mengidentifikasi tipe instans yang memiliki atribut tertentu.
- Armada EC2 menggunakan perlindungan harga untuk menentukan tipe instans yang akan dikecualikan.
- Armada EC2 menentukan kumpulan kapasitas dari mana ia akan mempertimbangkan untuk meluncurkan instans berdasarkan AWS Wilayah atau Zona Ketersediaan yang memiliki jenis instans yang cocok.

- Armada EC2 menerapkan strategi alokasi yang ditentukan untuk menentukan dari kolam kapasitas yang digunakan untuk meluncurkan instans.

Perhatikan bahwa pemilihan tipe instans berbasis atribut tidak memilih kolam kapasitas yang akan digunakan untuk menyediakan armada; hal tersebut adalah tugas strategi alokasi.

Jika Anda menentukan strategi alokasi, Armada EC2 akan meluncurkan instans sesuai dengan strategi alokasi yang ditentukan.

- Untuk Instans Spot, pemilihan tipe instans berbasis atribut mendukung strategi alokasi `price-capacity-optimized`, `capacity-optimized`, dan `lowest-price`. Perhatikan bahwa kami tidak merekomendasikan strategi alokasi `lowest-price` Spot karena memiliki risiko interupsi tertinggi untuk Instans Spot Anda.
- Untuk Instans Sesuai Permintaan, pemilihan tipe instans berbasis atribut mendukung strategi alokasi `lowest-price`.
- 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 Anda menggunakan tipe instans yang Anda anggap 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 mengecualikan jenis instans dengan harga 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 Instans On-Demand 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, seperti 999999.

Topik

- [Bagaimana jenis instans dengan harga terendah diidentifikasi](#)
- [Perlindungan harga Instans Sesuai Permintaan](#)
- [Perlindungan harga Spot Instance](#)
- [Tentukan ambang batas perlindungan harga](#)

Bagaimana jenis instans dengan harga terendah diidentifikasi

Amazon EC2 menentukan harga untuk mendasarkan 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 kali melihat jenis instance C, M, atau R generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, itu mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan terlihat jenis instance generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, itu mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan melihat jenis instance generasi sebelumnya yang cocok dengan atribut Anda, dan mengidentifikasi jenis instance dengan harga terendah.

Perlindungan harga Instans Sesuai Permintaan

Ambang batas perlindungan harga untuk jenis instans On-Demand dihitung sebagai persentase yang lebih tinggi daripada jenis instans On-Demand 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 instans On-Demand yang teridentifikasi adalah 0.4271, dan Anda tentukan 25, maka ambang harga 25% lebih tinggi dari 0.4271. Itu dihitung sebagai berikut: $0.4271 * 1.25 = 0.533875$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Sesuai Permintaan, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans On-Demand yang harganya lebih dari 0.533875

Perlindungan harga Spot Instance

Secara default, Amazon EC2 akan secara otomatis menerapkan perlindungan harga Instans Spot yang optimal untuk secara konsisten memilih 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 secara manual mengatur perlindungan harga, kami sarankan menggunakan opsi pertama.

- Persentase dari jenis instans On-Demand dengan harga terendah yang diidentifikasi []
`MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`

Misalnya, jika harga jenis instans On-Demand yang diidentifikasi adalah 0.4271 , dan Anda tentukan 60 , maka ambang harga adalah 60% dari 0.4271 . Itu dihitung sebagai berikut: $0.4271 * 0.60 = 0.25626$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans Spot apa pun yang harganya lebih dari 0.25626 .

- Persentase 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 tentukan 25 , maka ambang harga 25% lebih tinggi dari harga 0.1808 . Itu dihitung sebagai berikut: $0.1808 * 1.25 = 0.226$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans Spot apa pun yang harganya lebih dari 0.266 . Kami tidak menyarankan menggunakan parameter ini karena harga Spot dapat berfluktuasi, dan oleh karena itu ambang batas perlindungan harga Anda mungkin juga berfluktuasi.

Tentukan ambang batas perlindungan harga

Untuk menentukan ambang batas perlindungan harga

Saat membuat Armada EC2, konfigurasi armada untuk pemilihan tipe instans berbasis atribut, lalu lakukan hal berikut:

- Untuk menentukan ambang batas perlindungan harga Instans Sesuai Permintaan, dalam file konfigurasi JSON, dalam struktur `InstanceRequirements`, untuk

`OnDemandMaxPricePercentageOverLowestPrice`, masukkan ambang batas perlindungan harga sebagai persentase.

- Untuk menentukan ambang perlindungan harga Instans Spot, dalam file konfigurasi JSON, dalam `InstanceRequirements` struktur, tentukan salah satu parameter berikut:
 - Untuk `MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`, masukkan ambang perlindungan harga sebagai persentase.
 - Untuk `SpotMaxPricePercentageOverLowestPrice`, masukkan ambang perlindungan harga sebagai persentase.

Untuk informasi selengkapnya tentang cara membuat armada, lihat [Membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut](#).

Note

Saat membuat Armada EC2, jika Anda mengatur `TargetCapacityUnitType` ke `vcpu` atau `memory-mib`, ambang batas perlindungan harga akan diterapkan berdasarkan harga per vCPU atau per memori, bukan harga per instans.

Pertimbangan

- Anda dapat menentukan tipe instans atau atribut instans di Armada EC2, tetapi tidak dapat menentukan keduanya pada saat yang bersamaan.

Saat menggunakan CLI, penempatan templat peluncuran akan menimpa templat peluncuran. Misalnya, jika templat peluncuran berisi tipe instans dan penempatan 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 penempatan, 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 untuk menggunakan pemilihan tipe instans berbasis atribut menggunakan AWS CLI.

Untuk membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut (AWS CLI)

Gunakan perintah (AWS CLI) [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
```

Contoh file *file_name*.json

Contoh berikut berisi parameter yang mengonfigurasi Armada EC2 untuk menggunakan pemilihan tipe instans berbasis atribut, dan diikuti dengan penjelasan teks.

```
{  
  "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"  
}
```

Atribut untuk pemilihan tipe instans berbasis atribut ditentukan dalam struktur InstanceRequirements. Dalam contoh ini, dua atribut 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 kemungkinan atribut yang dapat Anda tentukan, lihat [InstanceRequirements](#) di Referensi API Amazon EC2.

Note

Jika InstanceRequirements disertakan dalam konfigurasi armada, InstanceType dan WeightedCapacity harus dikecualikan; keduanya tidak dapat menentukan konfigurasi armada pada saat yang sama sebagai atribut instans.

JSON juga berisi konfigurasi armada berikut:

- "AllocationStrategy": "*price-capacity-optimized*" – Strategi alokasi untuk Instans Spot di armada.
- "LaunchTemplateName": "*my-launch-template*", "Version": "*1*" – Templat peluncuran berisi beberapa informasi konfigurasi instans, tetapi jika ada tipe instans yang ditentukan, tipe instans tersebut akan diganti oleh atribut yang ditentukan dalam InstanceRequirements.
- "TotalTargetCapacity": *20* – Kapasitas target adalah 20 instans.
- "DefaultTargetCapacityType": "*spot*" – Kapasitas default adalah Instans Spot.
- "Type": "*instant*" – Tipe permintaan untuk armada adalah instant.

Contoh konfigurasi 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 penimpanan](#)
- [Konfigurasi yang valid: Template peluncuran tunggal dengan banyak InstanceRequirements](#)
- [Konfigurasi yang valid: Dua templat peluncuran, masing-masing dengan penimpanan](#)
- [Konfigurasi yang valid: Hanya InstanceRequirements yang ditentukan, tidak ada nilai atribut yang tumpang tindih](#)
- [Konfigurasi tidak valid: Overrides berisi InstanceRequirements dan InstanceType](#)
- [Konfigurasi tidak valid: Dua Overrides berisi InstanceRequirements dan InstanceType](#)
- [Konfigurasi tidak valid: Nilai atribut tumpang tindih](#)

Konfigurasi yang valid: Templat peluncuran tunggal dengan penimpanan

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 banyak `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 penimpanan

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: **Overrides** berisi **InstanceRequirements** dan **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 **Overrides** berisi **InstanceRequirements** dan **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"
    }
  }
}
```

Melihat pratinjau tipe instans dengan atribut tertentu

Anda dapat menggunakan AWS CLI 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 konfigurasi sebagai berikut:

Note

Anda harus memberikan nilai untuk `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount`, dan `MemoryMiB`. Anda dapat menghilangkan atribut lainnya; saat dihilangkan, nilai default digunakan.

Untuk penjelasan tentang setiap atribut dan nilai default-nya, lihat [get-instance-types-from-instance-requirements](#) dalam Referensi Baris Perintah Amazon EC2.

- 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 \
```



```
|| 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.

Mengonfigurasi Armada EC2 untuk pencadangan Sesuai Permintaan

Jika Anda memiliki kebutuhan penskalaan yang mendesak dan tidak dapat diprediksi, seperti situs web berita yang harus menskalakan selama peristiwa berita besar atau peluncuran game, sebaiknya tentukan tipe instans alternatif untuk Instans Sesuai Permintaan, jika opsi yang Anda pilih tidak memiliki kapasitas yang tersedia. Misalnya, Anda mungkin lebih memilih Instans Sesuai Permintaan `c5.2xlarge`, tetapi jika kapasitas yang tersedia tidak mencukupi, Anda bersedia menggunakan beberapa instans `c4.2xlarge` selama beban puncak. Dalam kasus ini, Armada EC2 berupaya memenuhi semua kapasitas target Anda menggunakan instans `c5.2xlarge`, tetapi jika kapasitas tidak mencukupi, Armada EC2 akan secara otomatis meluncurkan instans `c4.2xlarge` untuk memenuhi kapasitas target.

Topik

- [Memprioritaskan tipe instans untuk kapasitas Sesuai Permintaan](#)
- [Menggunakan Reservasi Kapasitas untuk Instans Sesuai Permintaan](#)

Memprioritaskan tipe instans untuk kapasitas Sesuai Permintaan

Saat Armada EC2 berupaya memenuhi kapasitas Sesuai Permintaan Anda, Armada EC2 secara default akan meluncurkan tipe instans dengan harga terendah terlebih dahulu. Jika `AllocationStrategy` diatur ke `prioritized`, Armada EC2 akan menggunakan prioritas untuk menentukan tipe instans yang akan digunakan pertama kali dalam memenuhi kapasitas Sesuai Permintaan. Prioritas ditetapkan ke penempatan templat peluncuran, dan prioritas tertinggi diluncurkan terlebih dahulu.

Contoh: Memprioritaskan tipe instans

Dalam contoh ini, Anda mengonfigurasi tiga penempatan templat peluncuran, masing-masing dengan tipe instans yang berbeda.

Harga Sesuai Permintaan untuk tipe instans beragam harganya. 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 akan memenuhi kapasitas Sesuai Permintaan dengan memulai dari tipe 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 penempatan templat peluncuran sehingga tipe instans digunakan dalam urutan prioritas, sebagai berikut:

- `m5.large` – prioritas 1
- `m4.large` – prioritas 2
- `m5a.large` – prioritas 3

Menggunakan Reservasi Kapasitas untuk Instans Sesuai Permintaan

Dengan Reservasi Kapasitas Sesuai Permintaan, Anda dapat memesan kapasitas komputasi untuk Instans Sesuai Permintaan di Zona Ketersediaan tertentu untuk durasi berapa pun. Anda dapat mengonfigurasi Armada EC2 untuk menggunakan Reservasi Kapasitas terlebih dahulu saat meluncurkan Instans Sesuai Permintaan.

Reservasi Kapasitas dikonfigurasi sebagai `open` atau `targeted`. Armada EC2 dapat meluncurkan Instans Sesuai Permintaan ke dalam Reservasi Kapasitas `open` atau `targeted`, sebagai berikut:

- Jika Reservasi Kapasitas adalah `open`, Instans Sesuai Permintaan yang memiliki atribut yang cocok secara otomatis akan berjalan dalam kapasitas terpesan.
- Jika Reservasi Kapasitas adalah `targeted`, Instans Sesuai Permintaan harus secara khusus menargetkannya untuk dijalankan dalam kapasitas terpesan. Hal ini berguna untuk menggunakan Reservasi Kapasitas tertentu atau untuk mengontrol kapan harus menggunakan Reservasi Kapasitas tertentu.

Jika Anda menggunakan Reservasi Kapasitas `targeted` di Armada EC2, harus ada Reservasi Kapasitas yang cukup untuk memenuhi kapasitas target Sesuai Permintaan, atau 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 cukup Reservasi Kapasitas; jika kehabisan Reservasi Kapasitas sebelum kapasitas target Sesuai Permintaan terpenuhi, armada dapat meluncurkan kapasitas target yang tersisa ke dalam kapasitas Sesuai Permintaan reguler.

Untuk menggunakan Reservasi Kapasitas dengan Armada EC2

1. Konfigurasi armada sebagai tipe `instant`. Anda tidak dapat menggunakan Reservasi Kapasitas untuk armada tipe lain.
2. Konfigurasi 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 untuk memenuhi kapasitas Sesuai Permintaan, jika armada menemukan bahwa lebih dari satu kolam instans memiliki Reservasi Kapasitas yang cocok yang tidak terpakai, armada akan menentukan kolam yang akan digunakan untuk meluncurkan Instans Sesuai Permintaan berdasarkan strategi alokasi Sesuai Permintaan (`lowest-price` atau `prioritized`).

Untuk contoh cara mengonfigurasi armada agar menggunakan Reservasi Kapasitas agar memenuhi kapasitas Sesuai Permintaan, lihat [Contoh konfigurasi Armada EC2](#), khususnya Contoh 5 hingga 7.

Untuk informasi tentang mengonfigurasi Reservasi Kapasitas, lihat [Reservasi Kapasitas Sesuai Permintaan](#) dan [FAQ Reservasi Kapasitas Sesuai Permintaan](#).

Penyeimbangan Ulang Kapasitas

Anda dapat mengonfigurasi Armada EC2 untuk meluncurkan Instans Spot pengganti jika Amazon EC2 memancarkan rekomendasi penyeimbangan ulang guna memberi tahu Anda bahwa Spot Instans memiliki risiko interupsi yang tinggi. Penyeimbangan Ulang Kapasitas membantu Anda mempertahankan ketersediaan beban kerja dengan secara proaktif menambah armada Anda dengan Instans Spot baru sebelum instans yang berjalan diinterupsi oleh Amazon EC2. Untuk informasi selengkapnya, lihat [Rekomendasi penyeimbangan ulang instans EC2](#).

Untuk mengonfigurasi Armada EC2 guna meluncurkan Instans Spot pengganti, gunakan perintah [create-fleet](#) (AWS CLI) dan parameter yang relevan di struktur `MaintenanceStrategies`. Untuk informasi selengkapnya, lihat [contoh konfigurasi peluncuran](#).

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

`ReplacementStrategy` untuk Armada EC2 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` yang dikombinasikan dengan `launch-before-terminate` untuk menghindari penggantian Instans Spot yang juga menaikkan risiko interupsi.

`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

Jika Anda mengonfigurasi Armada EC2 untuk Penyeimbangan Ulang Kapasitas, pertimbangkan hal berikut:

Berikan sebanyak mungkin kolam kapasitas Spot dalam permintaan

Konfigurasi Armada EC2 Anda untuk menggunakan lebih dari satu tipe 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

Instans Spot pengganti Anda mungkin berada dalam risiko tinggi mengalami interupsi jika Anda menggunakan strategi alokasi `lowest-price`. 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. Untuk menghindari peningkatan risiko gangguan, kami sangat menyarankan untuk tidak menggunakan strategi alokasi `lowest-price`, dan sebagai gantinya menyarankan strategi alokasi `capacity-optimized` atau `capacity-optimized-prioritized`. 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](#).

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 Ulang Kapasitas tidak diaktifkan, dan jika pemberitahuan interupsi Instans Spot akan diterima pada pukul 14:32 untuk instans-A, Anda hanya memiliki waktu hingga dua menit untuk mengambil tindakan, tetapi Instans-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

Ketika Armada EC2 dikonfigurasi untuk Penyeimbangan Ulang Kapasitas, armada tersebut berupaya meluncurkan Instans Spot pengganti yang baru untuk setiap Instans Spot yang menerima rekomendasi penyeimbangan ulang. 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 dikonfigurasi 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 EC2 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 penyeimbangan ulang, tetapi tidak ada lagi instans pengganti yang diluncurkan karena armada tidak dapat melebihi dua kali lipat dari kapasitas targetnya.

Perhatikan bahwa Anda dikenai biaya untuk semua instans saat berjalan.

Sebaiknya konfigurasi Armada EC2 untuk mengakhiri Instans Spot yang menerima rekomendasi penyeimbangan ulang

Jika Anda mengonfigurasi Armada EC2 untuk Penyeimbangan Ulang Kapasitas, sebaiknya pilih `launch-before-terminate` dengan penundaan pengakhiran yang sesuai hanya jika Anda dapat memprediksi berapa lama prosedur pematian instans Anda akan selesai. 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 EC2 tidak memperhitungkan instans yang menerima rekomendasi penyeimbangan ulang saat menghitung kapasitas yang terpenuhi saat menskalakan ke dalam atau ke luar

Jika Armada EC2 dikonfigurasi untuk Penyeimbangan Ulang Kapasitas, dan Anda mengubah kapasitas target untuk menskalakan ke dalam atau menskalakan ke luar, armada tidak akan memperhitungkan instans yang ditandai untuk penyeimbangan ulang 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 EC2 dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi penyeimbangan ulang, sehingga Amazon EC2 meluncurkan 10 instans pengganti baru, yang menghasilkan kapasitas 110 instans pengganti yang terpenuhi. 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 EC2 dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi penyeimbangan ulang, sehingga armada meluncurkan 10 instans pengganti baru, yang menghasilkan kapasitas 110 instans yang terpenuhi. 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.

Penimpaan harga maksimum

Setiap Armada EC2 dapat menyertakan harga maksimum global atau menggunakan harga default (harga Sesuai Permintaan). Armada menggunakan ini sebagai harga maksimum default untuk setiap spesifikasi peluncurannya.

Secara opsional, Anda dapat menentukan harga maksimum dalam satu atau beberapa spesifikasi peluncuran. Harga ini khusus untuk spesifikasi peluncuran. Jika spesifikasi peluncuran menyertakan harga tertentu, Armada EC2 akan menggunakan harga maksimum ini, sehingga menimpa harga maksimum global. Spesifikasi peluncuran lainnya yang tidak menyertakan harga maksimum tertentu tetap menggunakan harga maksimum global.

Kontrol pengeluaran

Armada EC2 berhenti meluncurkan instans ketika telah memenuhi salah satu parameter berikut: `TotalTargetCapacity` atau `MaxTotalPrice` (jumlah maksimum yang ingin Anda bayarkan). Guna mengontrol jumlah yang Anda bayarkan per jam untuk armada, Anda dapat menentukan `MaxTotalPrice`. Ketika harga total maksimum tercapai, Armada EC2 berhenti meluncurkan instans meskipun belum memenuhi kapasitas target.

Contoh berikut menunjukkan dua skenario berbeda. Yang pertama, Armada EC2 berhenti meluncurkan instans ketika telah memenuhi kapasitas target. Yang kedua, Armada EC2 berhenti meluncurkan instans ketika telah mencapai jumlah maksimum yang ingin Anda bayarkan (`MaxTotalPrice`).

Contoh: Menghentikan peluncuran instans saat kapasitas target tercapai

Dengan pertimbangan permintaan untuk Instans Sesuai Permintaan `m4.large`, jika:

- Harga Sesuai Permintaan: 0,10 USD per jam
- `OnDemandTargetCapacity`: 10
- `MaxTotalPrice`: 1,50 USD

Armada EC2 meluncurkan 10 Instans Sesuai Permintaan karena total 1,00 USD (10 instans x 0,10 USD) tidak melebihi `MaxTotalPrice` dari 1,50 USD untuk Instans Sesuai Permintaan.

Contoh: Menghentikan peluncuran instans ketika harga total maksimum tercapai

Dengan pertimbangan permintaan untuk Instans Sesuai Permintaan `m4.large`, jika:

- Harga Sesuai Permintaan: 0,10 USD per jam
- OnDemandTargetCapacity: 10
- MaxTotalPrice: 0,80 USD

Jika Armada EC2 meluncurkan kapasitas target Sesuai Permintaan (10 Instans Sesuai Permintaan), total biaya per jam adalah sebesar 1,00 USD. Ini lebih dari jumlah (0,80 USD) yang ditentukan untuk MaxTotalPrice untuk Instans Sesuai Permintaan. Untuk mencegah pengeluaran yang melebihi kesediaan Anda, Armada EC2 hanya meluncurkan 8 Instans Sesuai Permintaan (di bawah kapasitas target Sesuai Permintaan) karena meluncurkan lebih banyak akan melampaui MaxTotalPrice.

Pembobotan instans Armada EC2

Saat membuat Armada EC2, Anda dapat menentukan unit kapasitas yang akan dikontribusikan oleh setiap tipe instans untuk performa aplikasi. Anda kemudian dapat menyesuaikan harga maksimum untuk setiap spesifikasi peluncuran dengan menggunakan pembobotan instans.

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 EC2 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 menyertakan contoh penghitungan untuk menentukan harga per unit Armada EC2 dengan 10 kapasitas target.

Jenis instans	Bobot instans	Kapasitas target	Jumlah instans yang diluncurkan	Harga per jam instans	Harga per unit jam
r3.xlarge	2	10	5 (10 dibagi 2)	\$0,05	\$0,025 (,05 dibagi 2)

Jenis instans	Bobot instans	Kapasitas target	Jumlah instans yang diluncurkan	Harga per jam instans	Harga per unit jam
r3.8xlarge	8	10	2 (10 dibagi 8, hasil dibulatkan)	\$0,10	\$0,0125 (,10 dibagi 8)

Gunakan pembobotan instans Armada EC2 sebagai berikut untuk menyediakan kapasitas target yang Anda inginkan di kolam dengan harga terendah per unit pada saat pemenuhan:

1. Tetapkan kapasitas target untuk Armada EC2 baik dalam instans (default) atau dalam unit pilihan Anda, seperti CPU virtual, 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 EC2 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 untuk r3.2xlarge per jam instans dibagi 6), Armada EC2 akan meluncurkan empat instans (24 dibagi 6).

Jika spesifikasi peluncuran kedua memberikan harga terendah per unit (harga untuk c3.xlarge per jam instans dibagi 5), Armada EC2 akan meluncurkan lima instans ini (24 dibagi 5, hasil dibulatkan).

Pembobotan instans dan strategi alokasi

Pertimbangkan permintaan Armada EC2 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 EC2 akan meluncurkan empat instans (30 dibagi 8, hasilnya dibulatkan). 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.

Bekerja dengan Armada EC2

Untuk mulai menggunakan Armada EC2, Anda membuat permintaan yang menyertakan kapasitas target total, kapasitas Sesuai Permintaan, satu atau beberapa spesifikasi peluncuran untuk instans, dan harga maksimum yang ingin Anda bayarkan. Permintaan armada harus menyertakan templat peluncuran yang menentukan informasi yang dibutuhkan armada untuk meluncurkan instans, seperti AMI, tipe instans, subnet atau Zona Ketersediaan, dan satu atau beberapa grup keamanan. Anda dapat menentukan penempatan spesifikasi peluncuran untuk tipe instans, subnet, Zona Ketersediaan, dan harga maksimum yang ingin Anda bayarkan, dan Anda dapat menetapkan kapasitas tertimbang untuk setiap penempatan spesifikasi peluncuran.

Armada EC2 meluncurkan Instans Sesuai Permintaan ketika terdapat kapasitas yang tersedia, dan meluncurkan Instans Spot saat harga maksimum Anda melebihi harga Spot dan terdapat kapasitas yang tersedia.

Jika armada Anda menyertakan Instans Spot, Amazon EC2 dapat mencoba mempertahankan kapasitas target armada saat harga Spot berubah.

Tipe permintaan Armada EC2 `maintain` atau `request` tetap aktif sampai habis masa berlakunya atau Anda menghapusnya. Saat menghapus tipe armada `maintain` atau `request`, Anda dapat menentukan jika penghapusan mengakhiri instans dalam armada tersebut. Sebaliknya, Instans Sesuai Permintaan berjalan hingga Anda mengakhirinya, dan Instans Spot berjalan hingga diinterupsi atau Anda mengakhirinya.

Daftar Isi

- [Status permintaan Armada EC2](#)
- [Prasyarat Armada EC2](#)

- [Pemeriksaan kondisi Armada EC2](#)
- [Menghasilkan file konfigurasi JSON Armada EC2](#)
- [Membuat Armada EC2](#)
- [Menandai Armada EC2](#)
- [Jelaskan Armada EC2 Anda](#)
- [Memodifikasi Armada EC2](#)
- [Hapus Armada EC2](#)

Status permintaan Armada EC2

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. Permintaan tersebut dapat mencakup Instans Sesuai Permintaan, Instans Spot, atau keduanya. 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 Armada EC2 dihapus dan tidak meluncurkan instans 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 berada dalam status ini sampai semua instans diakhiri.

deleted

Armada EC2 dihapus dan tidak memiliki instans yang berjalan. Permintaan tersebut dihapus dua hari setelah instansnya diakhiri.

Prasyarat Armada EC2

Untuk membuat Armada EC2, prasyarat berikut harus ada:

- [Templat peluncuran](#)
- [Peran tertaut layanan untuk Armada EC2](#)
- [Memberikan akses ke kunci terkelola pelanggan untuk digunakan dengan AMI terenkripsi dan snapshot EBS](#)
- [Izin untuk pengguna Armada EC2](#)

Templat peluncuran

Templat peluncuran menyertakan informasi tentang instans yang akan diluncurkan, seperti tipe instans, Zona Ketersediaan, dan harga maksimum yang bersedia Anda bayarkan. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans dari templat peluncuran](#).

Peran tertaut layanan 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` – Mendeskripsikan Amazon Machine Image (AMI) untuk Instans Spot.
- `ec2:DescribeInstanceStatus` – Mendeskripsikan status Instans Spot.
- `ec2:DescribeSubnets` – Mendeskripsikan subnet untuk Instans Spot.

- `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, lalu pilih Buat peran.
3. Pada halaman Select type of trusted entity, lakukan hal berikut:
 - a. Untuk jenis 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. EC2 - Kasus penggunaan Armada akan secara otomatis membuat kebijakan dengan izin IAM yang diperlukan dan akan menyarankan `AWSServiceRoleForEC2Fleet` sebagai nama peran.

- c. Pilih Berikutnya.
4. Pada halaman Tambahkan izin, pilih Berikutnya.
 5. Pada halaman Nama, tinjau, dan buat, pilih Buat peran.

Jika Anda tidak perlu lagi menggunakan Armada EC2, kami sarankan Anda menghapus `AWSServiceRoleForEC2Fleet` peran tersebut. Setelah peran ini dihapus dari akun Anda, Anda dapat membuat peran tersebut kembali jika Anda membuat armada lain.

Untuk informasi selengkapnya, lihat [Menggunakan peran tertaut layanan](#) di Panduan Pengguna IAM.

Memberikan akses ke kunci terkelola 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 peran untuk menggunakan kunci AWSServiceRoleForEC2Fleet 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 terkelola pelanggan

- Gunakan perintah [create-grant](#) untuk menambahkan hibah ke kunci yang dikelola pelanggan dan untuk menentukan prinsipal (peran AWSServiceRoleForEC2Fleet terkait layanan) yang diberikan izin untuk melakukan operasi yang diizinkan hibah. Kunci yang dikelola pelanggan ditentukan oleh parameter `key-id` dan ARN kunci yang dikelola pelanggan. Prinsipal ditentukan oleh `grantee-principal` parameter dan ARN dari peran terkait AWSServiceRoleForEC2Fleet 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.

```
{
  "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, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:
 - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
 - (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Pemeriksaan kondisi Armada EC2

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 Anda](#).

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 Instans Sesuai Permintaan.
- 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. Ikuti langkah-langkah ini untuk membuat Armada EC2. Untuk informasi selengkapnya, lihat [Membuat Armada EC2](#).
2. Untuk mengonfigurasi armada guna mengganti Instans Spot yang tidak sehat, dalam file JSON, untuk `ReplaceUnhealthyInstances` masukkan `true`.

Menghasilkan file konfigurasi JSON Armada EC2

Untuk melihat daftar lengkap parameter konfigurasi Armada EC2, Anda dapat membuat file JSON. Untuk penjelasan tentang setiap parameter, lihat [create-fleet](#) dalam Referensi Perintah AWS CLI .

Untuk membuat file JSON dengan semua kemungkinan parameter Armada EC2 menggunakan baris perintah

- 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
```

Contoh output

```
{  
  "DryRun": true,  
  "ClientToken": "",  
  "SpotOptions": {  
    "AllocationStrategy": "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": ""
}
```

Membuat Armada EC2

Untuk membuat Armada EC2, Anda hanya perlu menentukan parameter berikut:

- `LaunchTemplateId` atau `LaunchTemplateName` - Menentukan templat peluncuran yang akan digunakan (yang berisi parameter untuk instans yang akan diluncurkan, seperti tipe instans, Zona Ketersediaan, dan harga maksimum yang bersedia Anda bayarkan)
- `TotalTargetCapacity` – Menentukan total kapasitas target untuk armada
- `DefaultTargetCapacityType` – Menentukan apakah opsi pembelian default adalah Sesuai Permintaan atau Spot

Anda dapat menentukan banyak spesifikasi peluncuran yang menempa templat peluncuran. Spesifikasi peluncuran dapat bervariasi berdasarkan tipe instans, Zona Ketersediaan, subnet, dan harga maksimum, serta dapat mencakup kapasitas tertimbang yang berbeda. Atau, Anda dapat menentukan atribut yang harus dimiliki instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut. Untuk informasi selengkapnya, lihat [Pemilihan tipe instans berbasis atribut untuk Armada EC2](#).

Jika Anda tidak menentukan parameter, armada akan menggunakan nilai default untuk parameter tersebut.

Tentukan parameter armada dalam file JSON. Untuk informasi selengkapnya, lihat [Menghasilkan file konfigurasi JSON Armada EC2](#).

Saat ini tidak ada dukungan konsol untuk membuat Armada EC2.

Untuk membuat Armada EC2 (AWS CLI)

- Gunakan perintah [create-fleet](#) (AWS CLI) untuk membuat Armada EC2 dan tentukan file JSON yang berisi parameter konfigurasi armada.

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

Untuk file konfigurasi contoh, lihat [Contoh konfigurasi Armada EC2](#).

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.

```
{
  "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
  },
  {
    "LaunchTemplateAndOverrides": {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
        "Version": "1"
      },
      "Overrides": {
        "InstanceType": "c4.large",
        "AvailabilityZone": "us-east-1a"
      }
    },
    "Lifecycle": "on-demand",
    "InstanceIds": [
      "i-5678901234abcdef0",
      "i-5432109876abcdef9"
    ]
  }
]
}

```

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": []
}
```

Menandai Armada EC2

Untuk membantu mengategorikan dan mengelola permintaan Armada EC2, Anda dapat menandainya dengan metadata kustom. Anda dapat menetapkan tanda untuk permintaan Armada EC2 saat membuatnya, 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 tanda hanya untuk permintaan armada, atau hanya untuk instans yang diluncurkan oleh armada, atau hanya untuk volume yang dilampirkan ke instans yang diluncurkan oleh armada, atau untuk ketiganya.

Note

Untuk tipe armada `instant`, Anda dapat menandai volume yang dilampirkan ke Instans Sesuai Permintaan dan Instans Spot. Untuk tipe armada `request` atau `maintain`, Anda hanya dapat menandai volume yang dilampirkan ke Instans Sesuai Permintaan.

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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "TagEC2FleetRequest",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2:CreateFleet"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

Saat ini kami tidak mendukung izin tingkat sumber daya untuk sumber daya `create-fleet`. Jika Anda menentukan `create-fleet` 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:CreateFleet"
  ]
}
```

```
  ],  
  "Resource": "arn:aws:ec2:us-east-1:111122223333:create-fleet/*"  
}
```

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
- (Tidak disarankan) Pasang 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 Armada EC2, instans, dan volume yang ada (AWS CLI)

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 Anda

Anda dapat menjelaskan konfigurasi Armada EC2, instans di Armada EC2, dan riwayat peristiwa Armada EC2.

Untuk menjelaskan Armada EC2 Anda (AWS CLI)

Gunakan perintah [describe-fleets](#) untuk menjelaskan Armada EC2 Anda.

```
aws ec2 describe-fleets
```

Important

Jika armada bertipe instant, Anda harus menentukan ID armada, jika tidak maka tidak akan muncul dalam respons. Sertakan `--fleet-ids` sebagai berikut:

```
aws ec2 describe-fleets --fleet-ids fleet-8a22eee4-f489-ab02-06b8-832a7EXAMPLE
```

Contoh output

```
{  
  "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"
    }
  }
]
}

```

Gunakan perintah [describe-fleet-instances](#) untuk menjelaskan instans Armada EC2 tertentu. Daftar instans yang sedang berjalan yang dikembalikan diperbarui secara berkala dan mungkin sudah lawas.

```
aws ec2 describe-fleet-instances --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Contoh output

```

{
  "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"
}

```

Gunakan perintah [describe-fleet-history](#) guna mendeskripsikan riwayat untuk Armada EC2 yang ditentukan untuk waktu yang ditentukan.

```

aws ec2 describe-fleet-history --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE --
start-time 2018-04-10T00:00:00Z

```

Contoh output

```

{
  "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": "2018-04-09T23:53:20.000Z"
}

```

Memodifikasi Armada EC2

Anda dapat memodifikasi Armada EC2 yang berada dalam status `submitted` atau `active`. Saat Anda memodifikasi armada, maka armada tersebut memasuki status `modifying`.

Anda hanya dapat memodifikasi Armada EC2 yang bertipe `maintain`. Anda tidak dapat memodifikasi Armada EC2 yang bertipe `request` atau `instant`.

Anda dapat memodifikasi parameter Armada EC2 berikut:

- `target-capacity-specification` – Meningkatkan atau menurunkan kapasitas target untuk `TotalTargetCapacity`, `OnDemandTargetCapacity`, dan `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` dan `termination`.

Saat Anda meningkatkan kapasitas target, Armada EC2 meluncurkan instans tambahan sesuai dengan opsi pembelian instans yang ditentukan untuk `DefaultTargetCapacityType`, yang merupakan Instans Sesuai Permintaan atau Instans Spot.

[Jika `yaspot`, Armada EC2 meluncurkan Instans Spot tambahan sesuai dengan strategi alokasi. `DefaultTargetCapacityType`](#)

Saat Anda menurunkan kapasitas target, Armada EC2 akan menghapus permintaan terbuka apa pun yang melebihi kapasitas target baru. Anda dapat meminta agar armada mengakhiri instans hingga ukuran armada mencapai kapasitas target yang baru. Jika strategi alokasinya adalah `lowest-price`, armada akan mengakhiri instans dengan harga per unit tertinggi. Jika strategi alokasinya adalah `diversified`, armada akan mengakhiri instans di seluruh kolam. Atau, Anda dapat meminta Armada EC2 untuk mempertahankan armada agar tetap dalam ukuran saat ini, tetapi tidak mengganti Instans Spot apa pun yang diinterupsi atau instans apa pun yang Anda akhiri secara manual.

Saat Armada EC2 mengakhiri Instans Spot karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan gangguan Instans Spot.

Untuk memodifikasi Armada EC2 (AWS CLI)

Gunakan perintah [`modify-fleet`](#) untuk memperbarui kapasitas target Armada EC2 tertentu.

```
aws ec2 modify-fleet \  
  --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --target-capacity-specification TotalTargetCapacity=20
```

Jika Anda menurunkan kapasitas target tetapi ingin mempertahankan armada pada ukuran saat ini, Anda dapat memodifikasi perintah sebelumnya seperti berikut.

```
aws ec2 modify-fleet \  
  --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --target-capacity-specification TotalTargetCapacity=10 \  
  --excess-capacity-termination-policy no-termination
```

Hapus Armada EC2

Jika tidak lagi membutuhkan Armada EC2, Anda dapat menghapusnya. Setelah Anda menghapus armada, semua permintaan Spot yang terkait dengan armada tersebut akan dibatalkan, sehingga tidak ada Instans Spot baru yang diluncurkan.

Saat menghapus Armada EC2, Anda juga harus menentukan apakah Anda ingin mengakhiri semua instansnya. Instans tersebut mencakup Instans Sesuai Permintaan 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.

Jika Anda menentukan bahwa instans harus diakhiri saat armada dihapus, armada memasuki status `deleted_terminating`. 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 tipe instant dalam satu permintaan.
- Anda dapat menghapus hingga 100 armada jenis `maintain` atau `request` dalam satu permintaan.
- Anda dapat menghapus hingga 125 armada dalam satu permintaan, 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.
- Hingga 1000 instans dapat diakhiri dalam satu permintaan untuk menghapus armada instant.

Untuk menghapus Armada EC2 dan mengakhiri instansnya (AWS CLI)

Gunakan perintah [delete-fleets](#) dan parameter `--terminate-instances` untuk menghapus Armada EC2 tertentu serta mengakhiri instans yang terkait.

```
aws ec2 delete-fleets \
  --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
  --terminate-instances
```

Contoh output

```
{
  "UnsuccessfulFleetDeletions": [],
  "SuccessfulFleetDeletions": [
    {
      "CurrentFleetState": "deleted_terminating",
      "PreviousFleetState": "active",
      "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
    }
  ]
}
```

```
}
```

Untuk menghapus Armada EC2 tanpa mengakhiri instansnya (AWS CLI)

Anda dapat memodifikasi perintah sebelumnya menggunakan parameter `--no-terminate-instances` untuk menghapus Armada EC2 tertentu tanpa mengakhiri instans terkaitnya.

 Note

`--no-terminate-instances` tidak didukung untuk armada instant.

```
aws ec2 delete-fleets \  
  --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --no-terminate-instances
```

Contoh output

```
{  
  "UnsuccessfulFleetDeletions": [],  
  "SuccessfulFleetDeletions": [  
    {  
      "CurrentFleetState": "deleted_running",  
      "PreviousFleetState": "active",  
      "FleetId": "fleet-4b8aaae8-dfb5-436d-a4c6-3dafa4c6b7dcEXAMPLE"  
    }  
  ]  
}
```

Memecahkan masalah saat armada gagal dihapus

Jika Armada EC2 gagal dihapus, `UnsuccessfulFleetDeletions` dalam output akan menampilkan ID 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: VvuncIxxj7Z_CPGNYXWqnuFV-
YjByeAU66Q9752NtQ-I3-qnDLWs6JLFd
KnSMMiq5s6cGqjjPtEDpsnGHzyHasFH0aRYJpaDVravoW25azn6KNkUQQLFwhJyujt2dtNCdduJfrqcFYAj1EiRMkfDht7
BhturzDK6A560Y2nDSUiMmAB1y9UNTqaZJ9SNe5sNxKMqZaqKtjRbk02RZu5V2vn9VMk6fm2aMVHbY9JhLvGypLcMuJtJ76
VPiU5v2s-
UgZ7h0p2yth6ysUdh10Ng6dBYu8_y_HtEI54invCj4CoK0qawqzMNe6rcmCQHvtCxtXsbkgyaEbcwmrm2m01-
EMhekLFZeJLr
DtY0pYcEl4_nWFX1wtQDCnNNcmxnJZAoJvb3VMDYpDTsxjQv1Px0DZuqWHS23YXWVyzgnLtHerf2o4lUhGBw17mXsS07k7
PT9vrHtQiILor5VVTsjSPWg7edj__1rsnXhwPSu8gI48ZLRGrPQqFq0RmK0_QIE8N8s6NWzCK4yoX-9gDcheur0GpkprPIC
</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.

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Sid": "DeleteFleetsAndTerminateInstances",
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteFleets"
        "ec2:TerminateInstances"
      ],
      "Resource": "*"
    }
  ]
}
```

Armada Spot

Armada Spot adalah set Instans Spot dan secara opsional Instans Sesuai Permintaan yang diluncurkan berdasarkan kriteria yang Anda tentukan. Armada Spot memilih kolam kapasitas Spot yang memenuhi kebutuhan Anda dan meluncurkan Instans Spot untuk memenuhi kapasitas target armada. Secara default, Armada Spot diatur untuk mempertahankan kapasitas target dengan meluncurkan instans pengganti setelah Instans Spot dalam armada diakhiri. Anda dapat mengirim Armada Spot sebagai permintaan satu kali, yang tidak bertahan setelah instans telah diakhiri. Anda dapat menyertakan permintaan Instans Sesuai Permintaan dalam permintaan Armada Spot.

Note

Jika Anda ingin menggunakan konsol untuk membuat armada yang menyertakan Instans Spot, sebaiknya gunakan grup Auto Scaling daripada Armada Spot. Untuk informasi selengkapnya, lihat [Grup Auto Scaling dengan banyak tipe instans dan opsi pembelian](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

Jika Anda ingin menggunakan AWS CLI untuk membuat armada yang menyertakan Instans Spot, sebaiknya gunakan grup Auto Scaling atau Armada EC2 daripada Armada Spot. [RequestSpotFleet](#) API, yang menjadi dasar Spot Fleet, adalah API lama tanpa investasi yang direncanakan.

Untuk informasi selengkapnya tentang API yang disarankan untuk digunakan, lihat [Metode permintaan Spot mana yang terbaik untuk digunakan?](#)

Topik

- [Tipe permintaan Armada Spot](#)
- [Strategi konfigurasi Armada Spot](#)
- [Bekerja dengan Armada Spot](#)
- [CloudWatch metrik untuk Spot Fleet](#)
- [Penskalaan otomatis untuk Armada Spot](#)

Tipe permintaan Armada Spot

Terdapat dua tipe permintaan Armada Spot:

request

Jika Anda mengonfigurasi tipe permintaan sebagai `request`, Armada Spot akan mengajukan permintaan satu kali asinkron untuk kapasitas yang Anda inginkan. Setelah itu, jika kapasitas berkurang karena interupsi Spot, armada tidak akan berupaya untuk mengisi Instans Spot, juga tidak akan mengirimkan permintaan dalam kolam kapasitas Spot alternatif jika kapasitas tidak tersedia.

maintain

Jika Anda mengonfigurasi tipe permintaan sebagai `maintain`, Armada Spot akan mengajukan permintaan asinkron untuk kapasitas yang Anda inginkan, dan mempertahankan kapasitas dengan secara otomatis mengisi ulang setiap Instans Spot yang terinterupsi.

Untuk menentukan tipe permintaan di konsol Amazon EC2, lakukan hal berikut saat membuat permintaan Armada Spot:

- Untuk membuat Armada Spot tipe `request`, kosongkan kotak centang Pertahankan kapasitas target.
- Untuk membuat Armada Spot tipe `maintain`, pilih kotak centang Pertahankan kapasitas target.

Untuk informasi selengkapnya, lihat [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).

Kedua tipe permintaan tersebut mendapatkan keuntungan dari strategi alokasi. Untuk informasi selengkapnya, lihat [Strategi alokasi untuk Instans Spot](#).

Strategi konfigurasi Armada Spot

Armada Spot adalah kumpulan, atau armada, Instans Spot, dan secara opsional Instans Sesuai Permintaan.

Armada Spot berupaya meluncurkan sejumlah Instans Spot dan Instans Sesuai Permintaan untuk memenuhi kapasitas target yang Anda tentukan dalam permintaan Armada Spot. Permintaan Instans Spot terpenuhi jika ada kapasitas yang tersedia dan harga maksimum yang Anda tentukan dalam permintaan melebihi harga Spot saat ini. Armada Spot tersebut juga berupaya mempertahankan armada kapasitas targetnya jika Instans Spot Anda terinterupsi.

Anda juga dapat menetapkan jumlah maksimum per jam yang ingin Anda bayarkan untuk armada, dan Armada Spot meluncurkan instans hingga mencapai jumlah maksimum. Saat jumlah maksimum

yang ingin Anda bayarkan tercapai, armada akan berhenti meluncurkan instans meskipun belum memenuhi kapasitas target.

Kolam kapasitas Spot adalah set instans EC2 yang tidak terpakai dengan tipe instans (misalnya, `m5.large`), sistem operasi, Zona Ketersediaan, dan platform jaringan yang sama. Saat membuat permintaan Armada Spot, Anda dapat menyertakan banyak spesifikasi peluncuran, yang bervariasi menurut tipe instans, AMI, Zona Ketersediaan, atau subnet. Armada Spot memilih kolam kapasitas Spot yang digunakan untuk memenuhi permintaan, berdasarkan spesifikasi peluncuran yang disertakan dalam permintaan Armada Spot, dan konfigurasi permintaan Armada Spot. Instans Spot berasal dari kolam yang dipilih.

Daftar Isi

- [Merencanakan permintaan Armada Spot](#)
- [Strategi alokasi untuk Instans Spot](#)
- [Pemilihan tipe instans berbasis atribut untuk Armada Spot](#)
- [Sesuai Permintaan di Armada Spot](#)
- [Penyeimbangan Ulang Kapasitas](#)
- [Penimpaan harga spot](#)
- [Kontrol pengeluaran](#)
- [Pembobotan instans Armada Spot](#)

Merencanakan permintaan Armada Spot

Sebelum Anda membuat permintaan Armada Spot, tinjau [Praktik Terbaik Spot](#). Gunakan praktik terbaik ini saat Anda merencanakan permintaan Armada Spot agar Anda dapat menyediakan tipe instans yang Anda inginkan dengan harga serendah mungkin. Kami juga menyarankan Anda untuk melakukan hal berikut:

- Menentukan apakah ingin membuat Armada Spot yang mengirimkan permintaan satu kali untuk kapasitas target yang diinginkan, atau yang mempertahankan kapasitas target dari waktu ke waktu.
- Tentukan tipe instans yang memenuhi kebutuhan aplikasi Anda.
- Tentukan kapasitas target untuk permintaan Armada Spot Anda. Anda dapat menetapkan kapasitas target dalam instans atau dalam unit kustom. Untuk informasi selengkapnya, lihat [Pembobotan instans Armada Spot](#).

- Tentukan berapa bagian dari kapasitas target Armada Spot yang harus menjadi kapasitas Sesuai Permintaan. Anda dapat menentukan 0 untuk kapasitas Sesuai Permintaan.
- Tentukan harga Anda per unit, jika Anda menggunakan pembobotan instans. Untuk menghitung harga per unit, bagi harga per jam instans dengan jumlah unit (atau bobot) yang diwakili oleh instans ini. Jika Anda tidak menggunakan pembobotan instans, harga default per unit adalah harga per jam instans.
- Tinjau opsi yang memungkinkan untuk permintaan Armada Spot Anda. Untuk informasi selengkapnya, lihat perintah [request-spot-fleet](#) dalam Referensi Perintah AWS CLI . Untuk contoh tambahan, lihat [Konfigurasi contoh Armada Spot](#).

Strategi alokasi untuk Instans Spot

Konfigurasi peluncuran Anda menentukan semua kemungkinan kolam kapasitas Spot (tipe instans dan Zona Ketersediaan) tempat Armada Spot dapat meluncurkan Instans Spot. Namun, saat meluncurkan instans, Armada Spot menggunakan strategi alokasi yang Anda tentukan untuk memilih kolam tertentu dari semua kemungkinan kolam Anda.

Note

(Hanya instans Linux) Jika Anda mengonfigurasi Instans Spot untuk diluncurkan dengan [AMD SEV-SNP](#) diaktifkan, Anda akan mengganti biaya penggunaan per jam tambahan yang setara dengan 10% dari tarif [per jam On-Demand untuk jenis instans yang dipilih](#). Jika strategi alokasi menggunakan harga sebagai input, Armada EC2 tidak termasuk biaya tambahan ini; hanya harga Spot yang digunakan.

Strategi alokasi

Anda dapat menentukan salah satu strategi alokasi berikut untuk Instans Spot:

`priceCapacityOptimized`(direkomendasikan)

Armada Spot mengidentifikasi kolam dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami yakini memiliki peluang interupsi terendah dalam waktu dekat. Armada Spot kemudian meminta Instans Spot dari harga terendah dari kolam ini.

Strategi alokasi `priceCapacityOptimized` adalah pilihan terbaik untuk sebagian besar beban kerja Spot, seperti aplikasi terkontainerisasi tanpa status, layanan mikro, aplikasi web, pekerjaan data dan analitik, serta pemrosesan batch.

`capacityOptimized`

Armada Spot mengidentifikasi kolam dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami yakini memiliki peluang interupsi terendah dalam waktu dekat. Anda dapat secara opsional menetapkan prioritas untuk setiap tipe instans dalam armada menggunakan `capacityOptimizedPrioritized`. Armada Spot mengoptimalkan kapasitas terlebih dahulu, tetapi mempertimbangkan prioritas tipe instans dengan upaya terbaik.

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 `capacityOptimized` secara otomatis meluncurkan Instans Spot ke dalam kolam yang paling tersedia dengan melihat data kapasitas waktu nyata dan memprediksi kolam 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 `capacityOptimized` dapat menurunkan biaya keseluruhan beban kerja Anda.

Atau, Anda dapat menggunakan strategi alokasi `capacityOptimizedPrioritized` dengan parameter prioritas untuk mengurutkan tipe instans dari prioritas tertinggi ke terendah. Anda dapat mengatur prioritas yang sama untuk tipe instans yang berbeda. Armada Spot akan mengoptimalkan kapasitas terlebih dahulu, tetapi akan mempertimbangkan prioritas tipe instans 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. Penggunaan prioritas hanya didukung jika armada Anda menggunakan templat peluncuran. Perhatikan bahwa ketika Anda menetapkan prioritas untuk `capacityOptimizedPrioritized`, prioritas yang sama akan diterapkan pada Instans Sesuai Permintaan jika `AllocationStrategy Sesuai Permintaan` diatur menjadi `prioritized`.

`diversified`

Instans Spot didistribusikan di semua kolam.

Memilih strategi alokasi yang tepat

Anda dapat mengoptimalkan armada untuk kasus penggunaan dengan memilih strategi alokasi Spot yang sesuai. Untuk kapasitas target Instans Sesuai Permintaan, Armada Spot selalu memilih jenis instans yang paling murah berdasarkan harga On-Demand publik, sambil mengikuti strategi alokasi—baik, atau `priceCapacityOptimized`—untuk Instans Spot. `capacityOptimized diversified`

Menyeimbangkan harga terendah dan ketersediaan kapasitas

Untuk menyeimbangkan kompromi antara kolam kapasitas Spot dengan harga terendah dan kolam kapasitas Spot dengan ketersediaan kapasitas tertinggi, sebaiknya gunakan strategi alokasi `priceCapacityOptimized`. Strategi ini membuat keputusan terkait kolam yang akan meminta Instans Spot dari berdasarkan harga kolam dan ketersediaan kapasitas Instans Spot di kolam tersebut. Hal ini berarti bahwa kami akan meminta Instans Spot dari kolam yang kami yakini memiliki kemungkinan interupsi paling rendah dalam waktu dekat, dengan tetap mempertimbangkan harga.

Jika armada Anda menjalankan beban kerja yang tangguh dan tanpa status, termasuk aplikasi terkontainerisasi, layanan mikro, aplikasi web, pekerjaan data dan analitik, serta pemrosesan batch, maka gunakan strategi alokasi `priceCapacityOptimized` untuk penghematan biaya yang optimal dan ketersediaan kapasitas.

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 operasi titik pemeriksaan, Anda membuat strategi alokasi `priceCapacityOptimized` cocok untuk beban kerja karena strategi ini mengalokasikan kapasitas dari kolam dengan harga terendah yang juga menawarkan tingkat interupsi Instans Spot yang rendah.

Untuk contoh konfigurasi yang menggunakan strategi alokasi `priceCapacityOptimized`, lihat [Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas](#).

Ketika beban kerja memiliki biaya interupsi yang tinggi

Anda dapat menggunakan strategi `capacityOptimized` secara opsional jika menjalankan beban kerja yang menggunakan tipe instans dengan harga yang sama, atau jika biaya interupsi sangat signifikan sehingga penghematan biaya apa pun tidak memadai jika dibandingkan dengan peningkatan marginal dalam interupsi. 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. Untuk contoh konfigurasi yang menggunakan

strategi alokasi `capacityOptimized`, lihat [Contoh 7: Konfigurasi Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti](#).

Ketika kemungkinan interupsi harus diminimalkan tetapi preferensi untuk tipe instans tertentu menjadi penting, Anda dapat mengekspresikan prioritas kolam Anda dengan menggunakan strategi alokasi `capacityOptimizedPrioritized`, lalu mengatur urutan tipe instans yang akan digunakan dari prioritas tertinggi ke terendah. Untuk contoh konfigurasi, lihat [Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas](#).

Perhatikan bahwa prioritas hanya didukung jika armada Anda menggunakan templat peluncuran. Perhatikan juga bahwa saat Anda menetapkan prioritas untuk `capacityOptimizedPrioritized`, prioritas yang sama juga diterapkan pada Instans Sesuai Permintaan Anda jika `AllocationStrategy Sesuai Permintaan` diatur ke `prioritized`.

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 `priceCapacityOptimized` untuk memaksimalkan penghematan biaya sekaligus tetap mempertimbangkan ketersediaan kapasitas.

Jika armada Anda besar atau berjalan untuk waktu yang lama

Jika armada Anda berjumlah besar atau berjalan untuk waktu yang lama, Anda dapat meningkatkan ketersediaan armada dengan mendistribusikan Instans Spot di banyak kolam menggunakan strategi `diversified`. Misalnya, jika Armada Spot Anda menentukan 10 kolam dan kapasitas target 100 instans, armada akan meluncurkan 10 Instans Spot di setiap kolam. 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 `diversified`, Armada Spot tidak meluncurkan Instans Spot ke dalam kolam mana pun dengan harga Spot yang sama atau lebih tinggi dari [harga Sesuai Permintaan](#).

Mempertahankan kapasitas target

Setelah Instans Spot diakhiri karena perubahan harga Spot atau kapasitas yang tersedia dari kolam kapasitas Spot, Armada Spot tipe `maintain` akan meluncurkan Instans Spot pengganti. Strategi alokasi menentukan kolam tempat instans pengganti diluncurkan, sebagai berikut:

- Jika strategi alokasinya adalah `priceCapacityOptimized`, armada akan meluncurkan instans pengganti di kolam yang memiliki ketersediaan kapasitas Instans Spot paling banyak sekaligus

juga mempertimbangkan harga dan mengidentifikasi kolam dengan harga terendah dengan ketersediaan kapasitas yang tinggi.

- Jika strategi alokasinya adalah `capacityOptimized`, armada akan meluncurkan instans pengganti di kolam yang memiliki ketersediaan kapasitas Instans Spot terbanyak.
- Jika strategi alokasinya adalah `diversified`, armada akan mendistribusikan Instans Spot pengganti di seluruh kolam yang tersisa.

Pemilihan tipe instans berbasis atribut untuk Armada Spot

Ketika membuat Armada Spot, Anda harus menentukan satu atau lebih tipe instans untuk mengonfigurasi Instans Sesuai Permintaan dan Instans Spot di 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 vCPU minimum dan maksimum yang diperlukan untuk instans Anda, dan Armada Spot akan meluncurkan instans menggunakan tipe instans yang tersedia yang memenuhi kebutuhan vCPU tersebut.

Pemilihan tipe instans berbasis atribut sangat ideal untuk beban kerja dan kerangka kerja yang fleksibel dalam menentukan tipe instans yang digunakan, seperti ketika menjalankan kontainer atau armada web, memproses big data, dan mengimplementasikan alat integrasi dan deployment berkelanjutan (CI/CD).

Keuntungan

Pemilihan tipe instans berbasis atribut memiliki keuntungan berikut:

- Mudah menggunakan jenis instans yang tepat — 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 Spot, 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 penyimpanan templat peluncuran.

- Penggunaan otomatis tipe instans baru — Saat Anda menentukan atribut instance daripada tipe instans, armada Anda dapat menggunakan tipe instance generasi yang lebih baru saat dirilis, “pemeriksaan masa depan” konfigurasi armada.
- Fleksibilitas tipe instans — Saat Anda menentukan atribut instance daripada tipe instans, Spot Fleet dapat memilih dari berbagai jenis instans untuk meluncurkan Instans Spot, yang mengikuti [praktik terbaik Spot dari fleksibilitas tipe instans](#).

Topik

- [Cara kerja pemilihan tipe instans berbasis atribut](#)
- [Perlindungan harga](#)
- [Pertimbangan](#)
- [Buat Armada Spot dengan pemilihan tipe instans berbasis atribut](#)
- [Contoh konfigurasi 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. Armada Spot akan meluncurkan instans pada tipe instans yang tersedia yang memiliki atribut instans yang ditentukan.

Topik

- [Tipe atribut instans](#)
- [Tempat mengonfigurasi pemilihan tipe instans berbasis atribut](#)
- [Cara Armada Spot menggunakan pemilihan tipe 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 instance.
- Memori — Minimum dan GiBs maksimum memori per instance.

- Penyimpanan lokal — Apakah akan menggunakan EBS atau volume penyimpanan instans untuk penyimpanan lokal.
- Kinerja burstable — Apakah akan menggunakan keluarga instans T, termasuk tipe T4G, T3a, T3, dan T2.

Untuk deskripsi setiap atribut dan nilai default, lihat [InstanceRequirements](#) di Referensi API Amazon EC2.

Tempat mengonfigurasi pemilihan tipe instans berbasis atribut

Bergantung pada apakah Anda menggunakan konsol atau konsol AWS CLI, Anda dapat menentukan atribut instance untuk pemilihan jenis instans berbasis atribut sebagai berikut:

Dalam konsol, Anda dapat menentukan atribut instans di salah satu atau semua komponen konfigurasi armada berikut ini:

- Dalam templat peluncuran, lalu referensikan templat peluncuran dalam permintaan armada
- Dalam permintaan armada

Di dalam AWS CLI, Anda dapat menentukan atribut instance dalam satu atau semua komponen konfigurasi armada berikut:

- Dalam templat peluncuran, dan 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, tipe instans yang berbeda dapat menggunakan prosesor berbasis x86 dan Arm.

- Dalam spesifikasi peluncuran

Cara Armada Spot menggunakan pemilihan tipe instans berbasis atribut saat menyediakan armada

Armada Spot menyediakan armada dengan cara berikut:

- Armada Spot mengidentifikasi tipe instans yang memiliki atribut tertentu.
- Armada Spot menggunakan perlindungan harga untuk menentukan tipe instans mana yang akan dikecualikan.

- Armada Spot menentukan kumpulan kapasitas dari mana ia akan mempertimbangkan untuk meluncurkan instans berdasarkan AWS Wilayah atau Zona Ketersediaan yang memiliki jenis instans yang cocok.
- Armada Spot menerapkan strategi alokasi yang ditentukan untuk menentukan dari kolam kapasitas yang digunakan untuk meluncurkan instans.

Perhatikan bahwa pemilihan tipe instans berbasis atribut tidak memilih kolam kapasitas yang akan digunakan untuk menyediakan armada; hal tersebut adalah tugas strategi alokasi. Mungkin terdapat tipe instans dalam jumlah besar dengan atribut yang ditentukan, dan beberapa di antaranya mungkin mahal.

Jika Anda menentukan strategi alokasi, Armada Spot akan meluncurkan instans sesuai dengan strategi alokasi yang ditentukan.

- Untuk Instans Spot, pemilihan jenis instans berbasis atribut mendukung strategi `capacityOptimizedPrioritized` dan `capacityOptimized` alokasi.
- Untuk Instans Sesuai Permintaan, pemilihan jenis instans berbasis atribut mendukung strategi `lowestPrice` alokasi, yang menjamin bahwa Armada Spot akan meluncurkan Instans Sesuai Permintaan dari kumpulan kapasitas yang paling murah.
- 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 Spot menggunakan tipe instans yang Anda anggap 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 mengecualikan jenis instans dengan harga 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 Instans On-Demand 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, seperti 999999.

Topik

- [Bagaimana jenis instans dengan harga terendah diidentifikasi](#)
- [Perlindungan harga Instans Sesuai Permintaan](#)
- [Perlindungan harga Spot Instance](#)
- [Tentukan ambang batas perlindungan harga](#)

Bagaimana jenis instans dengan harga terendah diidentifikasi

Amazon EC2 menentukan harga untuk mendasarkan 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 kali melihat jenis instance C, M, atau R generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, itu mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan terlihat jenis instance generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, itu mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan melihat jenis instance generasi sebelumnya yang cocok dengan atribut Anda, dan mengidentifikasi jenis instance dengan harga terendah.

Perlindungan harga Instans Sesuai Permintaan

Ambang batas perlindungan harga untuk jenis instans On-Demand dihitung sebagai persentase yang lebih tinggi daripada jenis instans On-Demand 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 instans On-Demand yang teridentifikasi adalah 0.4271 , dan Anda tentukan 25 , maka ambang harga 25% lebih tinggi dari 0.4271 . Itu dihitung sebagai berikut: $0.4271 * 1.25 = 0.533875$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Sesuai Permintaan, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans On-Demand yang harganya lebih dari 0.533875

Perlindungan harga Spot Instance

Secara default, Amazon EC2 akan secara otomatis menerapkan perlindungan harga Instans Spot yang optimal untuk secara konsisten memilih 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 secara manual mengatur perlindungan harga, kami sarankan menggunakan opsi pertama.

- Persentase dari jenis instans On-Demand dengan harga terendah yang diidentifikasi []
`MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`

Misalnya, jika harga jenis instans On-Demand yang diidentifikasi adalah 0.4271 , dan Anda tentukan 60 , maka ambang harga adalah 60% dari 0.4271 . Itu dihitung sebagai berikut: $0.4271 * 0.60 = 0.25626$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans Spot apa pun yang harganya lebih dari 0.25626

- Persentase 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 tentukan 25 , maka ambang harga 25% lebih tinggi dari harga 0.1808 . Itu dihitung sebagai berikut: $0.1808 * 1.25 = 0.226$. Harga yang dihitung adalah maksimum yang bersedia Anda bayarkan untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan jenis instans Spot apa pun yang harganya lebih dari 0.266 . Kami tidak menyarankan menggunakan parameter ini karena harga Spot dapat berfluktuasi, dan oleh karena itu ambang batas perlindungan harga Anda mungkin juga berfluktuasi.

Tentukan ambang batas perlindungan harga

Untuk menentukan ambang batas perlindungan harga

Saat membuat Armada Spot, konfigurasi armada untuk pemilihan tipe instans berbasis atribut, lalu lakukan hal berikut:

- **Konsol**

Untuk menentukan ambang perlindungan harga Instans Sesuai Permintaan, di bawah Atribut Instans tambahan, pilih Perlindungan harga sesuai permintaan, lalu pilih Tambahkan atribut. Untuk persentase perlindungan harga Sesuai Permintaan, masukkan ambang perlindungan harga sebagai persentase.

Untuk menentukan ambang batas perlindungan harga Instans Spot, di bawah Atribut instans tambahan, pilih Perlindungan harga Spot, lalu pilih Tambahkan atribut. Pilih parameter dan masukkan ambang perlindungan harga sebagai persentase.

- **AWS CLI**

Untuk menentukan ambang batas perlindungan harga Instans Sesuai Permintaan, dalam file konfigurasi JSON, dalam struktur `InstanceRequirements`, untuk `OnDemandMaxPricePercentageOverLowestPrice`, masukkan ambang batas perlindungan harga sebagai persentase.

Untuk menentukan ambang perlindungan harga Instans Spot, dalam file konfigurasi JSON, dalam `InstanceRequirements` struktur, tentukan salah satu parameter berikut:

- Untuk `MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`, masukkan ambang perlindungan harga sebagai persentase.
- Untuk `SpotMaxPricePercentageOverLowestPrice`, masukkan ambang perlindungan harga sebagai persentase.

Untuk informasi selengkapnya tentang cara membuat armada, lihat [Buat Armada Spot dengan pemilihan tipe instans berbasis atribut](#).

Note

Saat membuat Armada Spot, jika Anda mengatur tipe Total kapasitas target ke vCPU atau Memori (MiB) (konsol) atau `TargetCapacityUnitType` ke `vcpu` atau `memory-mib` (AWS

CLI), ambang batas perlindungan harga diterapkan berdasarkan harga per vCPU atau per memori, bukan harga per instans.

Pertimbangan

- Anda dapat menentukan tipe instans atau atribut instans di Armada Spot, tetapi tidak dapat menentukan keduanya pada saat yang 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.

Buat Armada Spot dengan pemilihan tipe instans berbasis atribut

Anda dapat mengonfigurasi armada untuk menggunakan pemilihan tipe instans berbasis atribut menggunakan konsol Amazon EC2 atau AWS CLI.

Topik

- [Membuat Armada Spot menggunakan konsol](#)
- [Buat Armada Spot menggunakan AWS CLI](#)

Membuat Armada Spot menggunakan konsol

Guna mengonfigurasi Armada Spot untuk pemilihan tipe instans berbasis atribut (konsol)

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 [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).

Saat membuat Armada Spot, konfigurasi 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.

Buat Armada Spot menggunakan AWS CLI

Untuk mengonfigurasi Armada Spot guna pemilihan tipe instans berbasis atribut (AWS CLI)

Gunakan perintah [request-spot-fleet](#) (AWS CLI) untuk membuat Armada Spot. Tentukan konfigurasi armada dalam file JSON.

```
aws ec2 request-spot-fleet \  
  --region us-east-1 \  
  --spot-fleet-request-config file://file_name.json
```

Contoh file *file_name*.json

Contoh berikut ini berisi parameter yang mengonfigurasi Armada Spot untuk menggunakan pemilihan tipe instans berbasis atribut, dan diikuti dengan penjelasan teks.

```
{  
  "AllocationStrategy": "priceCapacityOptimized",  
  "TargetCapacity": 20,  
  "Type": "request",  
  "LaunchTemplateConfigs": [{  
    "LaunchTemplateSpecification": {
```

```
"LaunchTemplateName": "my-launch-template",
"Version": "1"
},
"Overrides": [{
  "InstanceRequirements": {
    "VCpuCount": {
      "Min": 2
    },
    "MemoryMiB": {
      "Min": 4
    }
  }
}]
}]
}
```

Atribut untuk pemilihan tipe instans berbasis atribut ditentukan dalam struktur `InstanceRequirements`. Dalam contoh ini, dua atribut 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 kemungkinan atribut yang dapat Anda tentukan, lihat [InstanceRequirements](#) di Referensi API Amazon EC2.

Note

Jika `InstanceRequirements` disertakan dalam konfigurasi armada, `InstanceType` dan `WeightedCapacity` harus dikecualikan; keduanya tidak dapat menentukan konfigurasi armada pada saat yang sama sebagai atribut instans.

JSON juga berisi konfigurasi armada berikut:

- "AllocationStrategy": "*priceCapacityOptimized*" – Strategi alokasi untuk Instans Spot di armada.
- "LaunchTemplateName": "*my-launch-template*", "Version": "*1*" – Templat peluncuran berisi beberapa informasi konfigurasi instans, tetapi jika ada tipe instans yang ditentukan, tipe instans tersebut akan diganti oleh atribut yang ditentukan dalam InstanceRequirements.
- "TargetCapacity": *20* – Kapasitas target adalah 20 instans.
- "Type": "*request*" – Tipe permintaan untuk armada adalah request.

Contoh konfigurasi yang valid dan tidak valid

Jika Anda menggunakan AWS CLI untuk membuat Armada Spot, 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 banyak 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: Overrides berisi InstanceRequirements dan InstanceType](#)
- [Konfigurasi tidak valid: Dua Overrides berisi InstanceRequirements dan 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::000000000000: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 banyak `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::000000000000: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 penimpanan

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::000000000000: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::000000000000: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: **Overrides** berisi **InstanceRequirements** dan **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::000000000000: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 **Overrides** berisi **InstanceRequirements** dan **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::000000000000: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 kolom kapasitas ganda.

```

{
  "SpotFleetRequestConfig": {
    "AllocationStrategy": "priceCapacityOptimized",
    "ExcessCapacityTerminationPolicy": "default",
    "IamFleetRole": "arn:aws:iam::000000000000: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 AWS CLI 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 konfigurasi sebagai berikut:

Note

Anda harus memberikan nilai untuk `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount`, dan `MemoryMiB`. Anda dapat menghilangkan atribut lainnya; saat dihilangkan, nilai default digunakan.

Untuk penjelasan tentang setiap atribut dan nilai default-nya, lihat [get-instance-types-from-instance-requirements](#) dalam Referensi Baris Perintah Amazon EC2.

- 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           ||
|| c6a.xlarge           ||
...                     ||
```

4. Setelah mengidentifikasi tipe instans yang memenuhi kebutuhan Anda, catatlah atribut instans yang Anda gunakan sehingga Anda dapat menggunakannya saat mengonfigurasi permintaan armada.

Sesuai Permintaan di Armada Spot

Untuk memastikan bahwa Anda selalu memiliki kapasitas instans, Anda dapat menyertakan permintaan kapasitas Sesuai Permintaan dalam permintaan Armada Spot. Dalam permintaan Armada Spot, Anda menentukan kapasitas target yang diinginkan dan berapa banyak dari kapasitas tersebut yang harus Sesuai Permintaan. Saldo terdiri atas kapasitas Spot, yang diluncurkan jika tersedia kapasitas dan ketersediaan Amazon EC2. Misalnya, jika dalam permintaan Armada Spot Anda menentukan kapasitas target sebagai 10 dan kapasitas Sesuai Permintaan sebagai 8, Amazon EC2 akan meluncurkan 8 unit kapasitas sebagai Sesuai Permintaan, dan 2 unit kapasitas ($10-8=2$) sebagai Spot.

Memprioritaskan tipe instans untuk kapasitas Sesuai Permintaan

Ketika Armada Spot berupaya memenuhi kapasitas Sesuai Permintaan Anda, Armada Spot secara default akan meluncurkan tipe instans dengan harga terendah terlebih dahulu. Jika `OnDemandAllocationStrategy` diatur ke `prioritized`, Armada Spot akan menggunakan prioritas untuk menentukan tipe instans yang akan digunakan pertama kali dalam memenuhi kapasitas Sesuai Permintaan.

Prioritas ditetapkan ke penempatan templat peluncuran, dan prioritas tertinggi diluncurkan terlebih dahulu.

Contoh: Memprioritaskan tipe instans

Dalam contoh ini, Anda mengonfigurasi tiga penempatan templat peluncuran, masing-masing dengan tipe instans yang berbeda.

Harga Sesuai Permintaan untuk tipe instans beragam harganya. Berikut ini adalah tipe instans yang digunakan dalam contoh ini, yang disusun berdasarkan urutan harga, dimulai dengan tipe instans termurah:

- `m4.large` – termurah
- `m5.large`
- `m5a.large`

Jika Anda tidak menggunakan prioritas untuk menentukan urutan, armada akan memenuhi kapasitas Sesuai Permintaan dengan dimulai dari tipe instans termurah.

Namun, katakanlah Anda memiliki Instans Terpesan `m5.large` yang tidak terpakai yang ingin Anda gunakan terlebih dahulu. Anda dapat mengatur prioritas penempaan templat peluncuran sehingga tipe instans digunakan dalam urutan prioritas, sebagai berikut:

- `m5.large` – prioritas 1
- `m4.large` – prioritas 2
- `m5a.large` – prioritas 3

Penyeimbangan Ulang Kapasitas

Anda dapat mengonfigurasi Armada Spot untuk meluncurkan Instans Spot pengganti saat Amazon EC2 mengeluarkan rekomendasi penyeimbangan ulang untuk memberi tahu Anda bahwa Instans Spot memiliki risiko interupsi yang tinggi. Penyeimbangan Ulang Kapasitas membantu Anda mempertahankan ketersediaan beban kerja dengan secara proaktif menambah armada Anda dengan Instans Spot baru sebelum instans yang berjalan diinterupsi oleh Amazon EC2. Untuk informasi selengkapnya, lihat [Rekomendasi penyeimbangan ulang instans EC2](#).

Untuk mengonfigurasi Armada Spot guna meluncurkan Instans Spot pengganti, Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI.

- Konsol Amazon EC2: Anda harus memilih kotak centang Penyeimbangan ulang kapasitas saat Anda membuat Armada Spot. Untuk informasi selengkapnya, lihat langkah 6.d. di [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).
- AWS CLI: Gunakan perintah [request-spot-fleet](#) dan parameter yang relevan di struktur `SpotMaintenanceStrategies`. Untuk informasi selengkapnya, lihat [contoh konfigurasi peluncuran](#).

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

ReplacementStrategy untuk 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`.

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

Jika Anda mengonfigurasi Armada Spot untuk Penyeimbangan Ulang Kapasitas, pertimbangkan hal berikut:

Berikan sebanyak mungkin kolam kapasitas Spot dalam permintaan

Konfigurasikan Armada Spot Anda untuk menggunakan beberapa tipe 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 `capacityOptimized` atau `capacityOptimizedPrioritized` alokasi. 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](#).

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

pemhatian 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 pemhatian.*
- 14:30 – Rekomendasi penyeimbangan ulang diterima untuk instans-B, diganti dengan instans-C, sehingga memberi Anda waktu untuk memulai prosedur pemhatian.*
- 14:32 – Jika Penyeimbangan Ulang Kapasitas tidak diaktifkan, dan jika pemberitahuan interupsi Instans Spot akan diterima pada pukul 14:32 untuk instans-A, Anda hanya memiliki waktu hingga dua menit untuk mengambil tindakan, tetapi Instans-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 Spot dikonfigurasi untuk Penyeimbangan Ulang Kapasitas, Amazon EC2 berupaya meluncurkan Instans Spot pengganti baru untuk setiap Instans Spot yang menerima rekomendasi penyeimbangan ulang. 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 dikonfigurasi 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 Spot 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 penyeimbangan ulang, tetapi tidak ada lagi instans pengganti yang diluncurkan karena armada tidak dapat melebihi dua kali lipat dari kapasitas targetnya.

Perhatikan bahwa Anda dikenai biaya untuk semua instans saat berjalan.

Sebaiknya konfigurasi Armada Spot untuk mengakhiri Instans Spot yang menerima rekomendasi penyeimbangan ulang

Jika Anda mengonfigurasi Armada Spot untuk Penyeimbangan Ulang Kapasitas, sebaiknya pilih `launch-before-terminate` dengan penundaan pengakhiran yang sesuai hanya jika Anda dapat memprediksi berapa lama prosedur pematian instans Anda akan selesai. 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 Spot tidak menghitung instans yang menerima rekomendasi penyeimbangan ulang saat menghitung kapasitas yang terpenuhi selama menskalakan ke dalam atau ke luar

Jika Armada Spot Anda dikonfigurasi untuk Penyeimbangan Ulang Kapasitas, dan Anda mengubah kapasitas target untuk menskalakan ke dalam atau menskalakan ke luar, armada tidak akan memperhitungkan instans yang ditandai untuk penyeimbangan ulang 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 Spot dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi penyeimbangan ulang, sehingga Amazon EC2 meluncurkan 10 instans pengganti baru, yang menghasilkan kapasitas 110 instans pengganti yang terpenuhi. 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 Spot dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi penyeimbangan ulang, sehingga Amazon EC2 meluncurkan 10 instans pengganti baru, yang menghasilkan kapasitas 110 instans pengganti yang terpenuhi. 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.

Penimpanan harga spot

Setiap permintaan Armada Spot dapat menyertakan harga maksimum global, atau menggunakan harga default (harga Sesuai Permintaan). Armada Spot menggunakan ini sebagai harga maksimum default untuk setiap spesifikasi peluncurannya.

Secara opsional, Anda dapat menentukan harga maksimum dalam satu atau beberapa spesifikasi peluncuran. Harga ini khusus untuk spesifikasi peluncuran. Jika spesifikasi peluncuran menyertakan harga tertentu, Armada Spot akan menggunakan harga maksimum ini, sehingga menimpa harga maksimum global. Spesifikasi peluncuran lainnya yang tidak menyertakan harga maksimum tertentu tetap menggunakan harga maksimum global.

Kontrol pengeluaran

Armada Spot akan berhenti meluncurkan instans jika telah mencapai kapasitas target atau jumlah maksimum yang bersedia Anda bayarkan. Untuk mengontrol jumlah yang Anda bayarkan per jam untuk armada Anda, Anda dapat menentukan `SpotMaxTotalPrice` untuk Instans Spot dan `OnDemandMaxTotalPrice` untuk Instans Sesuai Permintaan. Jika total harga maksimum tercapai, Armada Spot akan berhenti meluncurkan instans meskipun belum memenuhi kapasitas target.

Contoh berikut menunjukkan dua skenario berbeda. Yang pertama, Armada Spot akan berhenti meluncurkan instans jika telah memenuhi kapasitas target. Yang kedua, Armada Spot akan berhenti meluncurkan instans jika telah mencapai jumlah maksimum yang bersedia Anda bayarkan.

Contoh: Menghentikan peluncuran instans saat kapasitas target tercapai

Dengan pertimbangan permintaan untuk Instans Sesuai Permintaan `m4.large`, jika:

- Harga Sesuai Permintaan: 0,10 USD per jam
- `OnDemandTargetCapacity`: 10
- `OnDemandMaxTotalPrice`: 1,50 USD

Armada Spot meluncurkan 10 Instans Sesuai Permintaan karena total 1,00 USD (10 instans x 0,10 USD) tidak melebihi `OnDemandMaxTotalPrice` sebesar 1,50 USD.

Contoh: Menghentikan peluncuran instans ketika harga total maksimum tercapai

Dengan pertimbangan permintaan untuk Instans Sesuai Permintaan `m4.large`, jika:

- Harga Sesuai Permintaan: 0,10 USD per jam
- `OnDemandTargetCapacity`: 10
- `OnDemandMaxTotalPrice`: 0,80 USD

Jika Armada Spot meluncurkan kapasitas target Sesuai Permintaan (10 Instans Sesuai Permintaan), total biaya per jam adalah 1,00 USD. Biaya ini lebih dari jumlah (0,80 USD) yang ditentukan untuk `OnDemandMaxTotalPrice`. Untuk mencegah pengeluaran yang melebihi kesediaan Anda, Armada Spot hanya meluncurkan 8 Instans Sesuai Permintaan (di bawah kapasitas target Sesuai Permintaan) karena meluncurkan lebih banyak akan melampaui `OnDemandMaxTotalPrice`.

Pembobotan instans Armada Spot

Ketika Anda meminta armada Instans Spot, Anda dapat menentukan unit kapasitas yang akan dikontribusikan oleh setiap tipe instans ke performa aplikasi, dan menyesuaikan harga maksimum untuk setiap kolam kapasitas Spot dengan menggunakan pembobotan instans.

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 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. Armada

Spot dapat memilih kolam mana pun yang Anda tentukan dalam spesifikasi peluncuran, meskipun kapasitas instans yang diluncurkan melebihi kapasitas target yang diminta.

Tabel berikut ini memberikan contoh perhitungan untuk menentukan harga per unit untuk permintaan Armada Spot dengan kapasitas target 10.

Jenis instans	Bobot instans	Harga per jam instans	Harga per unit jam	Jumlah instans yang diluncurkan
r3.xlarge	2	\$0,05	0,025 (,05 dibagi 2)	5 (10 dibagi 2)

Jenis instans	Bobot instans	Harga per jam instans	Harga per unit jam	Jumlah instans yang diluncurkan
r3.8xlarge	8	\$0,10	0,0125 (,10 dibagi 8)	2 (10 dibagi 8, hasil dibulatkan)

Gunakan pembobotan instans Armada Spot sebagai berikut untuk menyediakan kapasitas target yang Anda inginkan di kolam dengan harga terendah per unit pada saat pemenuhan:

1. Tetapkan kapasitas target untuk Armada Spot baik dalam instans (default) ataupun dalam unit pilihan Anda, seperti CPU virtual, memori, penyimpanan, atau throughput.
2. Tetapkan harga per unit.
3. Untuk setiap konfigurasi peluncuran, tentukan bobot, yang merupakan jumlah unit yang diwakili oleh tipe instans terhadap kapasitas target.

Contoh pembobotan instans

Pertimbangkan permintaan Armada Spot 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 untuk `r3.2xlarge` per jam instans dibagi 6), Armada Spot akan meluncurkan empat instans (24 dibagi 6).

Jika spesifikasi peluncuran kedua memberikan harga terendah per unit (harga untuk `c3.xlarge` per jam instans dibagi 5), Armada Spot akan meluncurkan lima instans ini (24 dibagi 5, hasil dibulatkan).

Pembobotan instans dan strategi alokasi

Pertimbangkan permintaan Armada Spot dengan konfigurasi berikut:

- Kapasitas target 30
- 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 Spot akan meluncurkan empat instans (30 dibagi 8, hasil dibulatkan). Dengan strategi `diversified`, Armada Spot meluncurkan satu instans di masing-masing dari tiga kolam, dan contoh keempat di kolam mana pun yang memberikan harga terendah per unit.

Bekerja dengan Armada Spot

Untuk mulai menggunakan Armada Spot, Anda membuat permintaan Armada Spot yang mencakup kapasitas target, porsi Sesuai Permintaan opsional, satu atau lebih spesifikasi peluncuran untuk instans, dan harga maksimum yang bersedia Anda bayarkan. Permintaan armada harus menyertakan spesifikasi peluncuran yang menentukan informasi yang dibutuhkan armada untuk meluncurkan instans, seperti AMI, tipe instans, subnet atau Zona Ketersediaan, dan satu atau lebih grup keamanan.

Jika armada Anda menyertakan Instans Spot, Amazon EC2 dapat mencoba mempertahankan kapasitas target armada saat harga Spot berubah.

Memodifikasi kapasitas target permintaan satu kali setelah dikirimkan tidak dimungkinkan. Untuk mengubah kapasitas target, batalkan permintaan dan kirimkan yang baru.

Permintaan Armada Spot tetap aktif hingga kedaluwarsa atau Anda membatalkannya. Saat Anda membatalkan permintaan armada, Anda dapat menentukan apakah membatalkan permintaan akan mengakhiri Instans Spot di armada tersebut.

Daftar Isi

- [Status permintaan Armada Spot](#)
- [Pemeriksaan kondisi Armada Spot](#)
- [Izin Armada Spot](#)
- [Membuat permintaan Armada Spot](#)
- [Menandai Armada Spot](#)
- [Menjelaskan Armada Spot Anda](#)
- [Memodifikasi permintaan Armada Spot](#)
- [Membatalkan permintaan Armada Spot](#)

Status permintaan Armada Spot

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 melampaui batas Armada Spot Anda, permintaan akan segera dibatalkan.
- `active` – Armada Spot telah divalidasi dan Amazon EC2 berupaya untuk mempertahankan jumlah target dari Instans Spot yang sedang berjalan. Permintaan tetap dalam keadaan ini sampai dimodifikasi atau dibatalkan.
- `modifying` – Permintaan Armada Spot sedang dimodifikasi. Permintaan tetap dalam status ini hingga modifikasi sepenuhnya diproses atau Armada Spot dibatalkan. `request` sekali pakai tidak dapat dimodifikasi, dan status ini tidak berlaku untuk permintaan Spot tersebut.
- `cancelled_running` – Armada Spot dibatalkan dan tidak meluncurkan Instans Spot tambahan. Instans Spot yang sudah ada akan terus berjalan hingga diinterupsi atau dihentikan. Permintaan tetap berada dalam status ini sampai semua instans diinterupsi atau diakhiri.
- `cancelled_terminating` – Armada Spot dibatalkan dan Instans Spot-nya berakhir. Permintaan tetap berada dalam status ini sampai semua instans diakhiri.
- `cancelled` – Armada Spot dibatalkan dan tidak memiliki Instans Spot yang sedang berjalan. Permintaan Armada Spot dihapus dua hari setelah instansnya diakhiri.

Pemeriksaan kondisi Armada Spot

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 Anda](#).

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 Instans Sesuai Permintaan.
- 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:DescribeInstanceState`.

Console

Untuk mengonfigurasi Armada Spot guna mengganti Instans Spot yang tidak sehat menggunakan konsol tersebut

1. Ikuti langkah-langkah ini untuk membuat Armada Spot. Untuk informasi selengkapnya, lihat [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).
2. Untuk mengonfigurasi armada agar mengganti Instans Spot yang tidak sehat, untuk Pemeriksaan kondisi, pilih Ganti instans yang tidak sehat. Untuk mengaktifkan opsi ini, Anda harus memilih Pertahankan kapasitas target terlebih dahulu.

AWS CLI

Untuk mengonfigurasi Armada Spot guna mengganti Instans Spot yang tidak sehat menggunakan AWS CLI

1. Ikuti langkah-langkah ini untuk membuat Armada Spot. Untuk informasi selengkapnya, lihat [Buat Armada Spot menggunakan AWS CLI](#).
2. Untuk mengonfigurasi armada guna mengganti Instans Spot yang tidak sehat, untuk `ReplaceUnhealthyInstances`, masukkan `true`.

Izin Armada Spot

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 sudah ada.

Gunakan petunjuk berikut untuk memberikan izin yang diperlukan dan membuat peran.

Izin dan peran

- [Memberikan izin kepada pengguna untuk Armada Spot](#)
- [Peran tertaut layanan untuk Armada Spot](#)
- [Peran terkait layanan 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.

```
{
  "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, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:
 - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
 - (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Peran tertaut layanan untuk Armada 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 Peran terkait layanan menyediakan cara aman untuk mendelegasikan izin ke AWS layanan karena hanya layanan tertaut yang dapat mengambil peran terkait layanan.

Untuk informasi selengkapnya, lihat [Menggunakan peran terkait layanan](#) dalam Panduan Pengguna IAM.

Amazon EC2 menggunakan peran terkait layanan yang diberi nama `AWSServiceRoleForEC2SpotFleet` untuk meluncurkan dan mengelola instans atas nama Anda.

⚠ Important

Jika Anda menentukan [AMI terenkripsi](#) atau snapshot Amazon EBS terenkripsi di Armada Spot, Anda harus memberikan `AWSServiceRoleForEC2SpotFleet` izin peran 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`

Amazon EC2 menggunakan `AWSServiceRoleForEC2SpotFleet` 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 `AWSServiceRoleForEC2SpotFleet` terkait layanan saat pertama kali Anda membuat Armada Spot menggunakan konsol.

Jika Anda memiliki permintaan Armada Spot aktif sebelum Oktober 2017, saat Amazon EC2 mulai mendukung peran terkait layanan ini, Amazon EC2 membuat peran tersebut di akun Anda. AWSServiceRoleForEC2SpotFleet 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 Armada Spot, Anda harus terlebih dahulu memastikan bahwa peran ini ada.

Untuk membuat AWSServiceRoleForEC2SpotFleet 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 jenis entitas Tepercaya, pilih AWS layanan.
 - b. Di bawah Kasus penggunaan, untuk Layanan atau kasus penggunaan, pilih EC2.
 - c. Untuk kasus Penggunaan, pilih EC2 - Armada Spot.
 - d. Pilih Berikutnya.
5. Pada halaman Tambahkan izin, pilih Berikutnya.
6. Pada halaman Nama, tinjau, dan buat, pilih Buat peran.

Untuk membuat AWSServiceRoleForEC2SpotFleet 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 Armada Spot, kami sarankan Anda menghapus AWSServiceRoleForEC2SpotFleetperan tersebut. 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 tertaut 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 Armada Spot dan Anda menggunakan kunci terkelola pelanggan untuk enkripsi, Anda harus

memberikan `AWSServiceRoleForEC2SpotFleet` izin peran 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 `AWSServiceRoleForEC2SpotFleet` terkait layanan) yang diberikan izin untuk melakukan operasi yang diizinkan hibah. CMK ditentukan oleh parameter `key-id` dan ARN CMK. Prinsipal ditentukan oleh `grantee-principal` parameter dan ARN dari peran terkait `AWSServiceRoleForEC2SpotFleet` 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/AWSServiceRoleForEC2SpotFleet \  
  --operations "Decrypt" "Encrypt" "GenerateDataKey" \  
  "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom" \  
  "ReEncryptTo"
```

Peran terkait layanan untuk Instans Spot

Amazon EC2 menggunakan peran terkait layanan yang diberi nama `AWSServiceRoleForEC2Spot` untuk meluncurkan dan mengelola Instans Spot atas nama Anda. Untuk informasi selengkapnya, lihat [Peran tertaut layanan untuk permintaan Instans 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 Armada Spot](#).

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 AWS layanan 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.

Pencegahan confused deputy lintas layanan

[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 `aws-ec2-spot-fleet-tagging-role` kepercayaan

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:account_id:spot-fleet-request/sfr-
*"
  },
  "StringEquals": {
    "aws:SourceAccount": "account_id"
  }
}
```

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:

```
{
  "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:account_id:spot-fleet-request/sfr-
*"
      },
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      }
    }
  }
}
}
}

```

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	Cakupan	<code>aws:SourceArn</code>
RequestSpotArmada	AWS STS (AssumeRole)	Batasi AssumeRole <code>aws-ec2-spot-fleet-tagging-role</code> kemampuan <code>spot-fleet-requests</code> di akun yang ditentukan.	<code>arn:aws:ec2:*:123456789012:spot-fleet-request/sfr-*</code>
RequestSpotArmada	AWS STS (AssumeRole)	Batasi AssumeRole <code>aws-ec2-spot-fleet-tagging-role</code> kemampuan <code>spot-fleet-requests</code> di akun yang ditentukan dan Wilayah yang ditentukan. Perhatikan	<code>arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-*</code>

Operasi API	Layanan yang dipanggil	Cakupan	aws:SourceArn
		bahwa peran ini tidak akan dapat digunakan di Wilayah lain.	
RequestSpotArmada	AWS STS (AssumeRole)	Batasi kemampuan AssumeRole di aws-ec2-spot-fleet-tagging-role hanya pada tindakan yang memengaruhi armada sfr-11111111-1111-1111-11111111 1111. Perhatikan bahwa peran ini mungkin tidak dapat digunakan untuk Armada Spot lainnya. Selain itu, peran ini tidak dapat digunakan untuk meluncurkan Armada Spot baru melalui request-spot-fleet.	arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-11111111-1111-1111-1111-1111-1111

Membuat permintaan Armada Spot

Menggunakan AWS Management Console, cepat membuat permintaan Armada Spot dengan memilih hanya aplikasi atau kebutuhan tugas dan spesifikasi komputasi minimum. Amazon EC2 mengonfigurasi armada yang paling sesuai dengan kebutuhan Anda dan mengikuti praktik terbaik Spot. Untuk informasi selengkapnya, lihat [Membuat permintaan Armada Spot dengan cepat \(konsol\)](#). Jika tidak, Anda dapat memodifikasi salah satu pengaturan default tersebut. Untuk informasi lebih lanjut, lihat [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#) dan [Buat Armada Spot menggunakan AWS CLI](#).

Opsi untuk membuat Armada Spot

- [Membuat permintaan Armada Spot dengan cepat \(konsol\)](#)
- [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#)
- [Buat Armada Spot menggunakan AWS CLI](#)

Membuat permintaan Armada Spot dengan cepat (konsol)

Ikuti langkah-langkah berikut untuk membuat permintaan Armada Spot dengan cepat.

Untuk membuat permintaan Armada Spot menggunakan pengaturan yang direkomendasikan (konsol)

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 Minta Instans Spot.
4. Di bawah Parameter peluncuran, pilih Konfigurasi 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 tipe unit, Anda dapat memilih Instans, vCPU, atau Memori (MiB).
7. Untuk Sekilas permintaan armada Anda, tinjau konfigurasi armada, dan pilih Luncurkan.

Buat permintaan Armada Spot menggunakan parameter yang ditentukan (konsol)

Anda dapat membuat Armada Spot menggunakan parameter yang Anda tentukan.

Untuk membuat permintaan Armada Spot menggunakan parameter yang ditentukan (konsol)

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 Minta Instans Spot.
4. Untuk Parameter peluncuran, lakukan hal berikut:
 - a. Untuk menentukan parameter peluncuran di konsol Spot, pilih Konfigurasi parameter peluncuran secara manual.

- b. Untuk AMI, pilih salah satu AMI dasar yang disediakan oleh AWS, atau pilih Cari AMI untuk menggunakan AMI dari komunitas pengguna kami, komunitas AWS Marketplace, atau salah satu milik Anda.

 Note

Jika AMI yang ditentukan dalam parameter peluncuran dideregistrasi atau dinonaktifkan, tidak ada instance baru yang dapat diluncurkan dari AMI. Untuk armada yang diatur untuk mempertahankan kapasitas target, kapasitas target tidak akan dipertahankan.

- c. (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.

- d. (Opsional) Perluas Parameter peluncuran tambahan, dan lakukan hal berikut:
 - i. (Opsional) Untuk mengaktifkan optimisasi Amazon EBS, untuk Dioptimalkan dengan EBS, pilih Luncurkan instans yang dioptimalkan EBS.
 - ii. (Opsional) Guna menambahkan penyimpanan tingkat blok sementara untuk instans Anda, untuk Penyimpanan instans, pilih Lampirkan saat peluncuran.
 - iii. (Opsional) Untuk menambahkan penyimpanan, pilih Tambahkan volume baru, dan tentukan volume penyimpanan instans tambahan atau volume Amazon EBS, tergantung pada tipe instans.
 - iv. (Opsional) Secara default, pemantauan dasar diaktifkan untuk instans Anda. Untuk mengaktifkan pemantauan terperinci, untuk Pemantauan, pilih Aktifkan pemantauan CloudWatch terperinci.
 - v. (Opsional) Guna menjalankan Instans Spot Khusus, untuk Penghunian, pilih Khusus - jalankan instans khusus.
 - vi. (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.
- vii. (Opsional) Agar instans Anda dapat dijangkau dari internet, untuk Menetapkan IP IPv4 Publik secara otomatis, pilih Aktifkan.
 - viii. (Opsional) Guna meluncurkan Instans Spot Anda dengan peran IAM, untuk Profil instans IAM, pilih peran tersebut.
 - ix. (Opsional) Untuk menjalankan skrip start-up, salin skrip tersebut ke Data pengguna.
 - x. (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.

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 Harga maksimum, Anda dapat menggunakan harga maksimum default (harga Sesuai Permintaan) atau menentukan harga maksimum yang ingin Anda bayarkan. Jika harga maksimum Anda lebih rendah daripada harga Spot untuk tipe instans yang Anda pilih, Instans Spot tidak akan diluncurkan.
 - d. (Opsional) Untuk membuat permintaan yang hanya berlaku selama jangka waktu tertentu, edit Permintaan berlaku mulai dan Permintaan berlaku sampai.
 - e. (Opsional) Secara default, kami mengakhiri Instans Spot saat permintaan Armada Spot kedaluwarsa. Agar Instans Spot tetap berjalan setelah permintaan Anda berakhir, hapus Akhiri instans saat permintaan kedaluwarsa.
 - f. (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 Unit komputasi minimum, pilih spesifikasi perangkat keras minimum (vCPU, memori, dan penyimpanan) yang Anda perlukan untuk aplikasi atau tugas, baik sebagai spesifikasi atau sebagai tipe instans.
 - Untuk sebagai spesifikasi, tentukan jumlah vCPU yang diperlukan dan jumlah memorinya.
 - Untuk sebagai tipe instans, terima tipe instans default, atau pilih Ubah tipe instans untuk memilih tipe instans yang berbeda.
7. 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 nantinya Anda dapat menambahkan kapasitas, pilih Pertahankan kapasitas target.
 - b. (Opsional) Untuk Sertakan kapasitas basis Sesuai Permintaan, tentukan jumlah unit Sesuai Permintaan 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 kapasitas Sesuai Permintaan opsional, Anda harus terlebih dahulu memilih templat peluncuran.

- c. (Opsional) Secara default, Amazon EC2 menghentikan Instans Spot saat terputus. Untuk mempertahankan kapasitas target, pilih Pertahankan kapasitas target. Anda kemudian dapat menentukan bahwa Amazon EC2 mengakhiri, menghentikan, atau hibernasi Instans Spot saat terputus. Untuk melakukannya, pilih opsi yang sesuai dari Perilaku interupsi.

 Note

Jika AMI yang ditentukan dalam parameter peluncuran dideregistrasi atau dinonaktifkan, tidak ada instance baru yang dapat diluncurkan dari AMI. Untuk armada yang diatur 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 hitungan detik) sebelum Armada Spot mengakhiri instans lama. Untuk informasi selengkapnya, lihat [Penyeimbangan Ulang Kapasitas](#).

- 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 [Kontrol pengeluaran](#).

8. 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 wizard dan segarkan daftar.

- b. (Opsional) Untuk Zona Ketersediaan, biarkan AWS 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 wizard dan segarkan daftar.

9. 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 [Pemilihan tipe instans berbasis atribut untuk Armada Spot](#).

- 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 tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.
 - ii. Untuk Memori (GiB), masukkan jumlah memori minimum dan maksimum yang diinginkan. Untuk menentukan tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.

- iii. (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 penjelasan tentang setiap atribut dan nilai default-nya, lihat [get-spot-placement-scores](#) dalam Referensi Baris Perintah Amazon EC2.
 - 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.
10. Untuk Strategi alokasi, pilih strategi yang memenuhi kebutuhan Anda. Untuk informasi selengkapnya, lihat [Strategi alokasi untuk Instans Spot](#).
 11. Untuk Sekilas permintaan armada Anda, tinjau konfigurasi armada dan lakukan penyesuaian apa pun jika perlu.
 12. (Opsional) Untuk mengunduh salinan konfigurasi peluncuran untuk digunakan dengan AWS CLI, pilih konfigurasi JSON.
 13. Pilih Luncurkan.

Tipe permintaan Armada Spot adalah `fleet`. Saat permintaan terpenuhi, permintaan tipe `instance` ditambahkan, di mana keadaannya `active` dan statusnya adalah `fulfilled`.

Buat Armada Spot menggunakan AWS CLI

Untuk membuat permintaan Armada Spot menggunakan AWS CLI

- Gunakan perintah [request-spot-fleet](#) untuk membuat permintaan Armada Spot.

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Untuk file konfigurasi contoh, lihat [Konfigurasi contoh Armada Spot](#).

Berikut adalah contoh output:

```
{  
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"  
}
```

Menandai Armada Spot

Untuk membantu mengategorikan dan mengelola permintaan Armada Spot, Anda dapat menandainya dengan metadata kustom. Anda dapat menetapkan tanda untuk permintaan Armada Spot saat Anda membuatnya, atau setelahnya. Anda dapat menetapkan tanda menggunakan konsol Amazon EC2 atau alat baris perintah.

Saat Anda menandai permintaan Armada Spot, instans dan volume yang diluncurkan oleh Armada Spot tidak secara otomatis ditandai. Anda perlu menandai instans dan volume yang diluncurkan oleh Armada Spot secara eksplisit. Anda dapat memilih untuk menetapkan tanda hanya untuk permintaan Armada Spot, atau hanya untuk instans yang diluncurkan oleh armada, atau hanya untuk volume yang dilampirkan ke instans yang diluncurkan oleh armada, atau ke ketiganya.

Note

Tanda volume hanya didukung untuk volume yang dilampirkan ke Instans Sesuai Permintaan. Anda tidak dapat menandai volume yang dilampirkan ke Instans Spot.

Untuk informasi selengkapnya tentang cara kerja tag, lihat [Tandai sumber daya Amazon EC2 Anda](#).

Daftar Isi

- [Prasyarat](#)
- [Menandai Armada Spot baru](#)
- [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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "TagSpotFleetRequest",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2:RequestSpotFleet"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

Saat ini kami tidak mendukung izin tingkat sumber daya untuk sumber daya `spot-fleet-request`. 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, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
- (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Menandai Armada Spot baru

Untuk menandai permintaan Armada Spot baru menggunakan konsol

1. Ikuti prosedur [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).
2. Untuk menambahkan tanda, perluas Konfigurasi tambahan, pilih Tambahkan tanda baru, dan masukkan kunci serta nilai untuk tanda tersebut. Ulangi untuk setiap tag.

Untuk setiap tanda, Anda dapat menandai permintaan Armada Spot dan instans dengan tanda yang sama. Untuk menandai keduanya, pastikan bahwa Tanda instans dan Tanda Armada telah dipilih. Untuk menandai permintaan Armada Spot saja, hapus Tanda instans. Untuk menandai instans yang diluncurkan oleh armada saja, hapus Tanda Armada.

3. Lengkapi bidang yang diperlukan untuk membuat permintaan Armada Spot, lalu pilih Luncurkan. Untuk informasi selengkapnya, lihat [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).

Untuk menandai permintaan Armada Spot baru menggunakan AWS CLI

Untuk menandai permintaan Armada Spot saat Anda membuatnya, konfigurasi konfigurasi permintaan Armada Spot sebagai berikut:

- 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.

Dalam contoh berikut, permintaan Armada Spot ditandai dengan dua tanda: Kunci=Lingkungan dan Nilai=Produksi, serta Kunci=Pusat-Biaya dan Nilai=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"
      }
    ],
    "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"
}
]
```

Menandai Armada Spot baru dan instans serta volume yang diluncurkannya

Untuk menandai permintaan Armada Spot baru dan 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, konfigurasi 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 Armada Spot ditandai dengan dua tanda: Kunci=Lingkungan dan Nilai=Produksi, serta Kunci=Pusat-Biaya dan Nilai=123. Instans yang diluncurkan oleh armada ditandai dengan satu tanda (yang sama dengan salah satu tanda untuk permintaan Armada Spot): Kunci=Pusat-Biaya dan Nilai=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, instans yang diluncurkan oleh armada ditandai dengan satu tanda: Kunci=Pusat-Biaya dan Nilai=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 Instans Sesuai Permintaan 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

Tanda volume hanya didukung untuk volume yang dilampirkan ke Instans Sesuai Permintaan. 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 Armada Spot, Anda dapat menambahkan tanda 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 Armada Spot 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 Armada Spot Anda

Armada Spot meluncurkan Instans Spot ketika harga maksimum Anda melebihi harga Spot dan kapasitas tersedia. Instans Spot berjalan hingga diinterupsi atau Anda mengakhirinya.

Untuk menjelaskan Armada Spot Anda (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. 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.

Untuk menjelaskan Armada Spot Anda (AWS CLI)

Gunakan perintah [describe-spot-fleet-requests](#) untuk menjelaskan permintaan Armada Spot Anda.

```
aws ec2 describe-spot-fleet-requests
```

Gunakan perintah [describe-spot-fleet-instances](#) guna menjelaskan Instans Spot untuk Armada Spot yang ditentukan.

```
aws ec2 describe-spot-fleet-instances \  
  --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Gunakan perintah [describe-spot-fleet-request-history](#) untuk menjelaskan riwayat permintaan Armada Spot yang ditentukan.

```
aws ec2 describe-spot-fleet-request-history \  
  --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --start-time 2015-05-18T00:00:00Z
```

Memodifikasi permintaan Armada Spot

Anda dapat memodifikasi permintaan Armada Spot yang aktif untuk menyelesaikan tugas berikut:

- Meningkatkan kapasitas target dan bagian Sesuai Permintaan
- Mengurangi kapasitas target dan bagian Sesuai Permintaan

Note

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.

Saat Anda meningkatkan kapasitas target, Armada Spot meluncurkan Instans Spot tambahan. Saat Anda meningkatkan bagian Sesuai Permintaan, Armada Spot meluncurkan Instans Sesuai Permintaan tambahan.

Saat Anda meningkatkan kapasitas target, Armada Spot meluncurkan Instans Spot tambahan sesuai dengan [strategi alokasi](#) untuk permintaan Armada Spot.

Saat Anda menurunkan kapasitas target, Armada Spot membatalkan permintaan terbuka apa pun 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.

Ketika Armada Spot mengakhiri instans karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan interupsi Instans Spot.

Untuk memodifikasi permintaan Armada Spot (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 Tindakan, Modifikasi kapasitas target.
5. Dalam Modifikasi kapasitas target, lakukan hal berikut:
 - a. Masukkan kapasitas target baru dan bagian Sesuai Permintaan.
 - b. (Opsional) Jika Anda menurunkan kapasitas target tetapi ingin mempertahankan armada pada ukurannya saat ini, hapus Akhiri instans.
 - c. Pilih Kirim.

Untuk mengubah permintaan Armada Spot menggunakan AWS CLI

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
```

Membatalkan permintaan Armada Spot

Jika Anda tidak lagi membutuhkan Armada Spot, Anda dapat membatalkan permintaan Armada Spot. 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. Instans tersebut mencakup Instans Sesuai Permintaan dan Instans Spot.

Jika Anda menentukan bahwa instans harus diakhiri saat permintaan armada dibatalkan, permintaan armada akan memasuki status `cancelled_terminating`. Jika tidak, permintaan armada akan masuk ke status `cancelled_running` dan instans tersebut terus berjalan hingga diinterupsi atau Anda mengakhirinya secara manual.

Pembatasan

- Anda dapat menghapus hingga 100 armada dalam satu permintaan. Jika Anda melebihi nomor yang ditentukan, tidak ada armada yang dihapus.

Untuk membatalkan permintaan Armada Spot (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 Tindakan, Batalkan permintaan.
5. Di kotak dialog Batalkan permintaan Spot, lakukan hal berikut:
 - a. Untuk mengakhiri instans yang terkait pada saat yang sama dengan membatalkan permintaan Armada Spot, biarkan kotak centang Akhiri instans dipilih. Untuk membatalkan permintaan Armada Spot tanpa mengakhiri instans terkait, kosongkan kotak centang Akhiri instans.
 - b. Pilih Konfirmasi.

Untuk membatalkan permintaan Armada Spot dan menghentikan instancenya menggunakan AWS CLI

Gunakan perintah [cancel-spot-fleet-requests](#) untuk membatalkan permintaan Armada Spot yang ditentukan dan mengakhiri Instans Sesuai Permintaan serta Instans Spot.

```
aws ec2 cancel-spot-fleet-requests \  
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
  --terminate-instances
```

Contoh output

```
{
  "SuccessfulFleetRequests": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "CurrentSpotFleetRequestState": "cancelled_terminating",
      "PreviousSpotFleetRequestState": "active"
    }
  ],
  "UnsuccessfulFleetRequests": []
}
```

Untuk membatalkan permintaan Armada Spot tanpa mengakhiri instans menggunakan AWS CLI

Anda dapat memodifikasi perintah sebelumnya menggunakan parameter `--no-terminate-instances` untuk membatalkan permintaan Armada Spot tertentu, tanpa mengakhiri Instans Sesuai Permintaan dan Instans Spot-nya.

```
aws ec2 cancel-spot-fleet-requests \
  --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
  --no-terminate-instances
```

Contoh Output

```
{
  "SuccessfulFleetRequests": [
    {
      "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
      "CurrentSpotFleetRequestState": "cancelled_running",
      "PreviousSpotFleetRequestState": "active"
    }
  ],
  "UnsuccessfulFleetRequests": []
}
```

CloudWatch metrik untuk Spot Fleet

Amazon EC2 menyediakan CloudWatch metrik Amazon yang dapat Anda gunakan untuk memantau Armada Spot Anda.

⚠ Important

Untuk memastikan keakuratannya, sebaiknya Anda mengaktifkan pemantauan mendetail saat menggunakan metrik ini. Untuk informasi selengkapnya, lihat [Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans Anda](#).

Untuk informasi selengkapnya tentang CloudWatch metrik yang disediakan oleh Amazon EC2, lihat [Pantau instans Anda menggunakan CloudWatch](#)

Metrik Armada Spot

AWS/EC2SpotNamespace menyertakan metrik berikut, ditambah CloudWatch metrik untuk Instans Spot di armada Anda. Untuk informasi selengkapnya, lihat [Metrik instans](#).

Metrik	Deskripsi
AvailableInstancePoolsCount	<p>Kolam kapasitas Spot yang ditentukan dalam permintaan Armada Spot.</p> <p>Unit: Jumlah</p>
BidsSubmittedForCapacity	<p>Kapasitas permintaan Armada Spot yang telah diajukan oleh Amazon EC2.</p> <p>Unit: Jumlah</p>
EligibleInstancePoolCount	<p>Kolam kapasitas Spot yang ditentukan dalam permintaan Armada Spot tempat Amazon EC2 dapat memenuhi permintaan. Amazon EC2 tidak memenuhi permintaan di kolam tempat harga maksimum yang bersedia Anda bayarkan untuk Instans Spot kurang dari harga Spot atau harga Spot lebih tinggi dari harga Instans Sesuai Permintaan.</p> <p>Unit: Jumlah</p>

Metrik	Deskripsi
FulfilledCapacity	<p>Kapasitas yang telah dipenuhi oleh Amazon EC2.</p> <p>Unit: Jumlah</p>
MaxPercentCapacityAllocation	<p>Nilai maksimum PercentCapacityAllocation di semua kolom Armada Spot yang ditentukan dalam permintaan Armada Spot.</p> <p>Unit: Persen</p>
PendingCapacity	<p>Perbedaan antara TargetCapacity dan Fulfilled Capacity .</p> <p>Unit: Jumlah</p>
PercentCapacityAllocation	<p>Kapasitas yang dialokasikan untuk kolom kapasitas Spot untuk dimensi tertentu. Agar nilai maksimum dapat tercatat di semua kolom kapasitas Spot, gunakan MaxPercentCapacityAllocation .</p> <p>Unit: Persen</p>
TargetCapacity	<p>Kapasitas target permintaan Armada Spot.</p> <p>Unit: Jumlah</p>
TerminatingCapacity	<p>Kapasitas yang sedang diakhiri karena kapasitas yang disediakan lebih besar dari kapasitas target.</p> <p>Unit: Jumlah</p>

Jika unit ukuran untuk metrik adalah Count, statistik yang paling berguna adalah Average.

Dimensi Armada Spot

Untuk memfilter data Armada Spot Anda, gunakan dimensi berikut.

Dimensi	Deskripsi
AvailabilityZone	Filter data berdasarkan Zona Ketersediaan.
FleetRequestId	Filter data berdasarkan permintaan Armada Spot.
InstanceType	Filter data menurut tipe instans.

Lihat CloudWatch metrik untuk Armada Spot Anda

Anda dapat melihat CloudWatch metrik untuk Armada Spot menggunakan CloudWatch konsol Amazon. Metrik ini ditampilkan sebagai grafik pemantauan. Grafik ini menunjukkan titik data jika Armada Spot aktif.

Metrik dikelompokkan terlebih dahulu berdasarkan namespace, kemudian berdasarkan berbagai kombinasi dimensi di dalam setiap namespace. Misalnya, Anda dapat menampilkan semua metrik Armada Spot atau grup metrik Armada Spot berdasarkan ID permintaan Armada Spot, tipe instans, atau Zona Ketersediaan.

Untuk menampilkan metrik Armada Spot

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi, pilih Metrik.
3. Pilih namespace Spot EC2.

Note

Jika namespace Spot EC2 tidak ditampilkan, ada dua alasan untuk ini. Entah Anda belum menggunakan Spot Fleet—hanya AWS layanan yang Anda gunakan mengirim metrik ke Amazon. CloudWatch Atau, jika Anda tidak menggunakan Armada Spot selama dua minggu terakhir, namespace tidak akan muncul.

4. (Opsional) Untuk memfilter metrik berdasarkan dimensi, pilih salah satu dari berikut ini:
 - Metrik Permintaan Armada – Dikelompokkan berdasarkan permintaan Armada Spot
 - Berdasarkan Zona Ketersediaan – Dikelompokkan berdasarkan permintaan Armada Spot dan Zona Ketersediaan
 - Berdasarkan Tipe Instans – Dikelompokkan berdasarkan permintaan Armada Spot dan tipe instans
 - Menurut Zona Ketersediaan/Tipe Instans – Dikelompokkan berdasarkan permintaan Armada Spot, Zona Ketersediaan, dan tipe instans
5. Untuk menampilkan data metrik, centang kotak di samping metrik.

EC2 Spot Search Metrics 1 to 18 of 18 Metrics

Fleet Request Metrics By Availability Zone By Instance Type By Availability Zone/Instance Type

Showing all results (18) for EC2 Spot > Fleet Request Metrics. For more results expand your search to All EC2 Spot Metrics. Select All | Clear

EC2 Spot > Fleet Request Metrics	
FleetRequestId	Metric Name
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	AvailableInstancePoolsCount
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	BidsSubmittedForCapacity
<input checked="" type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	CPUUtilization
<input type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	DiskReadBytes

Penskalaan otomatis untuk Armada Spot

Penskalaan otomatis adalah kemampuan untuk meningkatkan atau mengurangi kapasitas target Armada Spot Anda secara otomatis berdasarkan permintaan. Armada Spot dapat meluncurkan instans (menskalakan ke luar) atau mengakhiri instans (menskalakan ke dalam), dalam rentang yang Anda pilih, sebagai tanggapan terhadap satu atau beberapa kebijakan penskalaan.

Armada Spot mendukung tipe penskalaan otomatis berikut:

- [Penskalaan pelacakan target](#) – Meningkatkan atau menurunkan kapasitas armada saat ini berdasarkan nilai target untuk metrik tertentu. Hal tersebut mirip dengan cara termostat Anda menjaga suhu rumah—Anda memilih suhu dan termostat akan mengurus selebihnya.
- [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.

Jika Anda menggunakan [pembobotan instans](#), perlu diingat bahwa Armada Spot 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 $12 \times 0,2$ dari 12 untuk mendapatkan 9,6, kemudian membulatkannya menjadi 10, sehingga tidak ada tindakan penskalaan yang dilakukan.

Kebijakan penskalaan yang Anda buat untuk Armada Spot mendukung periode pendinginan. 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.

Sebaiknya skalakan 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 instans Anda ke CloudWatch dalam periode 1 menit, Anda harus secara khusus mengaktifkan pemantauan terperinci. Untuk informasi

lebih lanjut, lihat [Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans Anda](#) dan [Buat permintaan Armada Spot menggunakan parameter yang ditentukan \(konsol\)](#).

Untuk informasi selengkapnya tentang mengonfigurasi penskalaan pada Armada Spot, lihat sumber daya berikut:

- Bagian [application-autoscaling](#) dari Referensi Perintah AWS CLI
- [Referensi API Penskalaan Otomatis Aplikasi](#)
- [Panduan Pengguna Penskalaan Otomatis Aplikasi](#)

Izin IAM diperlukan untuk penskalaan otomatis Armada Spot

Penskalaan otomatis untuk Armada Spot dimungkinkan oleh kombinasi API Amazon EC2, CloudWatch Amazon, dan Application Auto Scaling. Permintaan Armada Spot dibuat dengan Amazon EC2, alarm dibuat dengan CloudWatch, dan kebijakan penskalaan dibuat dengan Application Auto Scaling.

Selain [izin IAM untuk Armada Spot](#) dan Amazon EC2, pengguna yang mengakses pengaturan penskalaan armada harus memiliki izin yang sesuai untuk layanan yang mendukung penskalaan dinamis. Pengguna harus memiliki izin untuk menggunakan tindakan yang ditunjukkan dalam contoh kebijakan berikut.

```
{
  "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 [Autentikasi dan Kontrol Akses](#) di Panduan Pengguna Penskalaan Otomatis Aplikasi.

Layanan Application Auto Scaling juga memerlukan izin untuk menjelaskan Armada Spot dan CloudWatch alarm Anda, dan izin untuk mengubah kapasitas target Armada Spot Anda 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 [Peran Tertaut Layanan](#) di Panduan Pengguna Penskalaan Otomatis Aplikasi.

Menskalakan Armada Spot menggunakan kebijakan pelacakan target

Dengan kebijakan penskalaan pelacakan target, Anda memilih metrik dan menetapkan nilai target. Spot Fleet membuat dan mengelola CloudWatch alarm yang memicu kebijakan penskalaan dan menghitung penyesuaian penskalaan berdasarkan metrik dan nilai target. Kebijakan penskalaan menambah atau menghapus kapasitas yang diperlukan untuk menjaga metrik berada pada, atau mendekati, nilai target yang ditentukan. Selain menjaga metrik agar mendekati nilai target, kebijakan penskalaan pelacakan target juga menyesuaikan dengan fluktuasi metrik karena pola muatan yang berfluktuasi dan meminimalkan fluktuasi cepat dalam kapasitas armada.

Anda dapat membuat lebih dari satu kebijakan penskalaan pelacakan target untuk Armada Spot, asalkan masing-masing menggunakan metrik yang berbeda. Armada diskalakan berdasarkan

kebijakan yang menyediakan kapasitas armada terbesar. Hal ini memungkinkan Anda untuk mencakup berbagai skenario dan memastikan bahwa selalu ada kapasitas yang cukup untuk memproses 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 karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan interupsi Instans Spot.

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.

Batasan

Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`.

Untuk mengonfigurasi kebijakan pelacakan target (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 Auto Scaling.
4. Jika penskalaan otomatis tidak dikonfigurasi, pilih Konfigurasikan.
5. 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.
6. Untuk Nama kebijakan, masukkan nama untuk kebijakan tersebut.
7. Pilih Metrik target.
8. Masukkan Nilai target untuk metrik.
9. Untuk Periode pendinginan, tentukan nilai baru (dalam detik) atau simpan default.
10. (Opsional) Pilih Nonaktifkan penskalaan ke dalam untuk menghilangkan pembuatan kebijakan penskalaan ke dalam berdasarkan konfigurasi saat ini. Anda dapat membuat kebijakan penskalaan ke dalam menggunakan konfigurasi yang berbeda.
11. 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](#).

Skalakan Armada Spot menggunakan kebijakan penskalaan bertahap

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 karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan interupsi Instans Spot.

Batasan

Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`, atau blok Spot.

Prasyarat

- 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 pengumpulan CloudWatch metrik jika layanan yang menyediakan metrik tidak mengaktifkannya secara default.

Untuk membuat CloudWatch alarm

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi, pilih Alarm.
3. Pilih Buat alarm.
4. Di halaman Tentukan metrik dan kondisi, pilih Pilih metrik.
5. Pilih EC2 Spot, Metrik Permintaan Armada, 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 Periode, 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 [Mengevaluasi 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 menangani data yang hilang](#) di CloudWatch Panduan 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. Pada Tambahkan deskripsi, Anda harus memasukkan nama serta deskripsi untuk alarm Anda dan pilih Berikutnya.
14. Pilih Buat alarm.

Untuk mengonfigurasi kebijakan penskalaan langkah terhadap Armada Spot (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 Auto Scaling.
4. Jika penskalaan otomatis tidak dikonfigurasi, pilih Konfigurasikan.
5. 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.
6. Untuk kebijakan penskalaan, Tipe kebijakan, pilih Kebijakan penskalaan langkah.
7. 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.
8. 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 `Add`, ketik `30` di bidang berikutnya, lalu pilih `percent`. 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.
9. Pilih `Simpan`.

Untuk mengonfigurasi kebijakan penskalaan langkah untuk Armada Spot Anda 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](#).

Menskalakan Armada Spot menggunakan penskalaan terjadwal

Penskalaan berdasarkan jadwal memungkinkan Anda menskalakan aplikasi sebagai respons terhadap perubahan permintaan yang dapat diprediksi. Untuk menggunakan penskalaan terjadwal, Anda membuat tindakan terjadwal, yang memberi tahu Armada Spot untuk melakukan aktivitas penskalaan pada waktu tertentu. Saat Anda membuat tindakan terjadwal, Anda menentukan Armada Spot, waktu aktivitas penskalaan harus dilakukan, kapasitas minimum, dan kapasitas maksimum. Anda dapat membuat tindakan terjadwal yang menskalakan satu kali saja atau menskalakan berdasarkan jadwal berulang.

Anda hanya dapat membuat tindakan terjadwal untuk Armada Spot yang sudah ada. Anda tidak dapat membuat tindakan terjadwal pada saat yang sama ketika Anda membuat Armada Spot.

Batasan

Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`, atau blok Spot.

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 dan pilih tab Penskalaan Terjadwal di dekat layar bagian bawah.
4. Pilih Buat Tindakan Terjadwal.
5. Untuk Nama, tentukan nama untuk tindakan terjadwal.
6. Masukkan nilai untuk Kapasitas minimum, Kapasitas maksimum, atau keduanya.
7. Untuk Perulangan, pilih Sekali.
8. (Opsional) Pilih tanggal dan waktu untuk Waktu mulai, Waktu berakhir, atau keduanya.
9. Pilih Kirim.

Untuk menskalakan pada jadwal yang 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 dan pilih tab Penskalaan Terjadwal di dekat layar bagian bawah.
4. 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 Pengguna CloudWatch Acara Amazon](#).
5. (Opsional) Pilih tanggal dan waktu untuk Waktu mulai, Waktu berakhir, atau keduanya.
6. 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 dan pilih tab Penskalaan Terjadwal di dekat layar bagian bawah.
4. Pilih tindakan terjadwal dan pilih Tindakan, Edit.
5. 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 dan pilih tab Penskalaan Terjadwal di dekat layar bagian bawah.
4. Pilih tindakan terjadwal dan pilih Tindakan, Hapus.
5. 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 peristiwa armada menggunakan Amazon EventBridge

Saat status Armada EC2 atau Armada Spot berubah, armada memancarkan notifikasi.

Pemberitahuan dibuat tersedia sebagai acara yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon CloudWatch Events). Peristiwa dipancarkan atas dasar upaya terbaik.

Dengan Amazon EventBridge, Anda dapat membuat aturan yang memicu tindakan terprogram sebagai respons terhadap suatu peristiwa. Misalnya, Anda dapat membuat dua EventBridge aturan, satu yang dipicu saat status armada berubah, dan yang dipicu saat instance dalam armada dihentikan. Anda dapat mengonfigurasi aturan pertama sehingga, jika status armada berubah, aturan tersebut akan menginvokasi topik SNS untuk mengirimkan notifikasi email kepada Anda. Anda dapat mengonfigurasi aturan kedua sehingga, jika sebuah instans diakhiri, aturan tersebut akan menginvokasi fungsi Lambda untuk meluncurkan instans baru.

Topik

- [Tipe peristiwa Armada EC2](#)
- [Tipe peristiwa Armada Spot](#)
- [Buat EventBridge aturan Amazon](#)

Tipe peristiwa Armada EC2

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.

Terdapat lima tipe peristiwa Armada EC2. Untuk setiap jenis acara, ada beberapa sub-jenis.

Acara dikirim ke EventBridge dalam format JSON. Bidang berikut dalam acara membentuk pola acara yang ditentukan dalam aturan, dan yang memicu tindakan:

```
"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.

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 saat Armada EC2 mengubah 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-6b0809bffff0a"
  ],
  "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

Armada EC2 mengirimkan `EC2 Fleet Spot Instance Request Change` peristiwa 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

Armada EC2 mengirimkan EC2 Fleet Instance Change acara ke Amazon EventBridge saat instance di 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

Armada EC2 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

Armada EC2 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

Ada lima tipe peristiwa Armada Spot. Untuk setiap tipe peristiwa, ada beberapa subtype.

Acara dikirim ke EventBridge dalam format JSON. Bidang dalam peristiwa berikut membentuk pola peristiwa yang ditentukan dalam aturan dan yang memicu tindakan:

```
"source": "aws.ec2spotfleet"
```

Mengidentifikasi bahwa peristiwa tersebut berasal dari Armada Spot.

```
"detail-type": "EC2 Spot Fleet State Change"
```

Mengidentifikasi tipe peristiwa.

```
"detail": { "sub-type": "submitted" }
```

Mengidentifikasi subtype peristiwa.

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

Armada Spot mengirimkan EC2 Spot Fleet State Change acara ke Amazon EventBridge saat Armada Spot mengubah 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

Armada Spot mengirimkan EC2 Spot Fleet Spot Instance Request Change peristiwa 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

Armada Spot 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

Armada Spot mengirimkan 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

Armada Spot 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.

Buat EventBridge aturan Amazon

Saat pemberitahuan perubahan status dipancarkan untuk Armada EC2 atau Armada Spot, acara untuk notifikasi 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.

Anda dapat menulis EventBridge aturan dan mengotomatiskan tindakan apa yang harus diambil ketika pola acara cocok dengan aturan.

Topik

- [Membuat EventBridge aturan Amazon untuk memantau peristiwa Armada EC2](#)
- [Membuat EventBridge aturan Amazon untuk memantau peristiwa Spot Fleet](#)

Membuat EventBridge aturan Amazon untuk memantau peristiwa Armada EC2

Ketika pemberitahuan perubahan status dipancarkan untuk Armada EC2, acara untuk pemberitahuan dikirim ke Amazon EventBridge dalam bentuk file JSON. Anda dapat menulis EventBridge aturan untuk mengotomatiskan tindakan apa yang harus diambil ketika pola peristiwa cocok dengan aturan. Jika EventBridge mendeteksi pola peristiwa yang cocok dengan pola yang ditentukan dalam aturan, EventBridge memanggil target (atau target) yang ditentukan dalam aturan.

Bidang berikut 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 sub tipe peristiwa.

Untuk daftar peristiwa Armada EC2 dan contoh data peristiwa, lihat [the section called “Tipe peristiwa Armada EC2”](#).

Contoh

- [Buat EventBridge aturan untuk mengirim pemberitahuan](#)
- [Buat EventBridge aturan untuk memicu fungsi Lambda](#)

Buat EventBridge aturan untuk mengirim pemberitahuan

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.

Sebelum membuat EventBridge aturan, Anda harus membuat topik Amazon SNS untuk email, pesan teks, atau notifikasi 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. 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 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 pesan application-to-person \(A2P\) di Panduan Pengembangan Layanan Pemberitahuan Sederhana Amazon](#).
 - 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 peristiwa 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 Armada EC2 saat instance diluncurkan. Sinyal dalam contoh ini dipancarkan sebagai peristiwa EC2 Fleet Instance Change, subtype 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, konfigurasi kodenya, lalu pilih Buat fungsi.

Untuk informasi selengkapnya tentang menggunakan Lambda, lihat [Membuat fungsi Lambda dengan konsol](#) dalam Panduan Developer AWS Lambda .

Untuk membuat EventBridge aturan untuk memicu fungsi Lambda saat instance di Armada EC2 mengubah 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. 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 sesuai dengan peristiwa EC2 Fleet Instance Change dan subtype 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: Log Status Instans Amazon EC2 Menggunakan dalam Panduan Pengembang](#). EventBridge AWS Lambda

Membuat EventBridge aturan Amazon untuk memantau peristiwa Spot Fleet

Ketika pemberitahuan perubahan status dipancarkan untuk Armada Spot, acara untuk notifikasi dikirim ke Amazon EventBridge dalam bentuk file JSON. Anda dapat menulis EventBridge aturan untuk mengotomatiskan tindakan apa yang harus diambil ketika pola peristiwa cocok dengan aturan. Jika EventBridge mendeteksi pola peristiwa yang cocok dengan pola yang ditentukan dalam aturan, EventBridge memanggil target (atau target) yang ditentukan dalam aturan.

Bidang berikut membentuk pola peristiwa yang ditentukan dalam aturan:

```
"source": "aws.ec2spotfleet"
```

Mengidentifikasi bahwa peristiwa tersebut berasal dari Armada Spot.

```
"detail-type": "EC2 Spot Fleet State Change"
```

Mengidentifikasi tipe peristiwa.

```
"detail": { "sub-type": "submitted" }
```

Mengidentifikasi subtype peristiwa.

Untuk daftar peristiwa Armada Spot dan contoh data peristiwa, lihat [the section called "Tipe peristiwa Armada Spot"](#).

Contoh

- [Buat EventBridge aturan untuk mengirim pemberitahuan](#)
- [Buat EventBridge aturan untuk memicu fungsi Lambda](#)

Buat EventBridge aturan untuk mengirim pemberitahuan

Contoh berikut membuat EventBridge aturan untuk mengirim email, pesan teks, atau pemberitahuan push seluler setiap kali Amazon EC2 memancarkan pemberitahuan perubahan status Armada Spot. Sinyal dalam contoh ini dipancarkan sebagai peristiwa EC2 Spot Fleet State Change, yang memicu tindakan yang ditentukan oleh aturan. Sebelum membuat EventBridge aturan, Anda harus membuat topik Amazon SNS untuk email, pesan teks, atau notifikasi push seluler.

Untuk membuat EventBridge aturan untuk mengirim pemberitahuan saat status Armada Spot 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. 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 sesuai dengan peristiwa EC2 Spot Fleet Instance Change.

```
{
  "source": ["aws.ec2spotfleet"],
  "detail-type": ["EC2 Spot 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 Spot EC2.
 - D. Untuk Tipe peristiwa, pilih Perubahan Instans Armada Spot 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 pesan application-to-person \(A2P\) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).
 - 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 peristiwa 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 Spot Fleet saat instance diluncurkan. Sinyal dalam contoh ini dipancarkan sebagai peristiwa EC2 Spot Fleet Instance Change, subtype 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, konfigurasi kodenya, lalu pilih Buat fungsi.

Untuk informasi selengkapnya tentang menggunakan Lambda, lihat [Membuat fungsi Lambda dengan konsol](#) dalam Panduan Developer AWS Lambda .

Untuk membuat EventBridge aturan untuk memicu fungsi Lambda saat instance di Armada Spot mengubah 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. 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 sesuai dengan peristiwa EC2 Spot Fleet Instance Change dan sub tipe launched.

```
{
  "source": ["aws.ec2spotfleet"],
  "detail-type": ["EC2 Spot 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 Spot EC2.
 - D. Untuk Tipe peristiwa, pilih Perubahan Instans Armada Spot 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: Log Status Instans Amazon EC2 Menggunakan dalam Panduan Pengembang](#). EventBridge AWS Lambda

Tutorial untuk Armada EC2 dan Armada Spot

Tutorial berikut ini memandu Anda melalui proses umum untuk membuat Armada EC2 dan Armada Spot.

Tutorial

- [Tutorial: Menggunakan Armada EC2 dengan pembobotan instans](#)
- [Tutorial: Menggunakan Armada EC2 dengan Sesuai Permintaan sebagai kapasitas primer](#)
- [Tutorial: Meluncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas yang ditargetkan](#)
- [Tutorial: Meluncurkan instans ke Blok Kapasitas](#)
- [Tutorial: Menggunakan Armada Spot dengan pembobotan instans](#)

Tutorial: Menggunakan Armada EC2 dengan 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 pembobotan instance, kapasitas target dapat sama dengan sejumlah instance (default) atau kombinasi faktor seperti inti (vCPU), memori (), dan penyimpanan (GBGiBs). 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 kapasitas target permintaan Armada EC2 mereka menjadi 20.

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 [harga Sesuai Permintaan](#) per jam instans sebagai titik awal untuk harga mereka. 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:

Jenis instans	Harga Sesuai Permintaan	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 [Membuat templat peluncuran](#).

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: Menggunakan Armada EC2 dengan Sesuai Permintaan sebagai kapasitas primer

Tutorial ini menggunakan perusahaan fiktif bernama ABC Online untuk menggambarkan proses permintaan Armada EC2 dengan Sesuai Permintaan sebagai kapasitas primer, dan kapasitas Spot jika tersedia.

Tujuan

ABC Online, perusahaan pengiriman restoran, ingin dapat menyediakan kapasitas Amazon EC2 di seluruh tipe instans EC2 dan opsi pembelian untuk mencapai skala, performa, dan biaya yang diinginkan.

Rencana

ABC Online membutuhkan kapasitas tetap untuk beroperasi selama periode puncak, tetapi ingin mendapatkan keuntungan dari peningkatan kapasitas dengan harga lebih rendah. ABC Online menentukan kebutuhan berikut untuk Armada EC2 mereka:

- Kapasitas Instans Sesuai Permintaan – ABC Online memerlukan 15 Instans Sesuai Permintaan untuk memastikan bahwa mereka dapat mengakomodasi lalu lintas pada periode puncak.
- Kapasitas Instans Spot – ABC Online ingin meningkatkan performa, tetapi dengan harga yang lebih rendah, dengan 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 [Membuat templat peluncuran](#).

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 kapasitas Sesuai Permintaan selalu terpenuhi, sementara keseimbangan kapasitas target dipenuhi sebagai Spot jika terdapat kapasitas dan ketersediaan.

Tutorial: Meluncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas yang ditargetkan

Tutorial ini memandu Anda melalui semua langkah yang harus Anda lakukan agar Armada EC2 Anda meluncurkan Instans Sesuai Permintaan ke Reservasi Kapasitas `targeted`.

Anda akan mempelajari cara mengonfigurasi armada untuk menggunakan Reservasi Kapasitas Sesuai Permintaan `targeted` terlebih dahulu saat meluncurkan Instans Sesuai Permintaan. Anda juga akan mempelajari cara mengonfigurasi armada sehingga saat total kapasitas target Sesuai Permintaan melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia, armada tersebut akan menggunakan strategi alokasi yang ditentukan untuk memilih kolam instans untuk meluncurkan kapasitas target yang tersisa.

Konfigurasi Armada EC2

Dalam tutorial ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 10 Instans Sesuai Permintaan
- Total Reservasi Kapasitas `targeted` yang tidak terpakai: 6 (kurang dari kapasitas target Sesuai Permintaan armada sebesar 10 Instans Sesuai Permintaan)
- Jumlah kolam Reservasi Kapasitas: 2 (`us-east-1a` dan `us-east-1b`)
- Jumlah Reservasi Kapasitas per kolam: 3
- Strategi alokasi Sesuai Permintaan: `lowest-price` (Jika jumlah Reservasi Kapasitas yang tidak terpakai kurang dari kapasitas target Sesuai Permintaan, armada akan menentukan kolam tempat meluncurkan kapasitas Sesuai Permintaan yang tersisa berdasarkan strategi alokasi Sesuai Permintaan.)

Perhatikan bahwa Anda juga dapat menggunakan strategi alokasi `prioritized` alih-alih strategi alokasi `lowest-price`.

Untuk meluncurkan Instans Sesuai Permintaan ke Reservasi Kapasitas `targeted`, Anda harus menjalankan 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 [Menggunakan Reservasi Kapasitas untuk Instans Sesuai Permintaan](#).

```
aws resource-groups create-group \  
  --name my-cr-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 Reservasi Kapasitas Sesuai Permintaan berdasarkan ARN.

```
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](#) untuk mendeskripsikan templat peluncuran secara opsional guna 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 kolom instans, masing-masing dengan tipe instans yang sama (`c5.xlarge`), tetapi dengan Zona Ketersediaan (`us-east-1a` dan `us-east-1b`) yang berbeda. Harga kolom 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 alokasi Sesuai Permintaan 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 Instans Sesuai Permintaan sebagai berikut:
 - 3 Instans Sesuai Permintaan diluncurkan ke dalam 3 Reservasi Kapasitas `c5.xlarge` targeted di `us-east-1a`
 - 3 Instans Sesuai Permintaan diluncurkan ke dalam 3 Reservasi Kapasitas `c5.xlarge` targeted di `us-east-1b`
- Untuk memenuhi kapasitas target, 4 Instans Sesuai Permintaan tambahan diluncurkan ke kapasitas Sesuai Permintaan reguler sesuai dengan strategi alokasi Sesuai Permintaan, yaitu

lowest-price dalam contoh ini. Namun, karena kolam memiliki harga yang sama (karena harganya adalah per Wilayah dan bukan per Zona Ketersediaan), armada meluncurkan 4 Instans Sesuai Permintaan yang tersisa ke salah satu kolam.

(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: 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. Untuk informasi selengkapnya tentang Blok Kapasitas, lihat [Blok Kapasitas untuk ML](#).

Anda dapat menggunakan Armada EC2 tipe instan untuk meluncurkan instans ke Blok Kapasitas. Untuk informasi selengkapnya, lihat [Gunakan Armada EC2 tipe 'instan'](#).

Dalam sebagian besar 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 akan menerima Pengecualian Kapasitas Tidak Mencukupi untuk kapasitas yang melebihi batas reservasi Blok Kapasitas.

Note

Untuk Blok Kapasitas, Armada EC2 tidak akan melakukan fallback guna meluncurkan Instans Sesuai Permintaan 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 mengajukan permintaan ke Armada EC2 untuk diluncurkan ke Blok Kapasitas yang belum dimulai, Anda akan menerima Kesalahan Kapasitas Tidak Cukup.

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 hingga Anda menghentikan atau mengakhirinya melalui panggilan API Amazon EC2 terpisah atau hingga Amazon EC2 mengakhiri instans saat reservasi Blok Kapasitas berakhir.

Pertimbangan

- Lebih dari satu Blok Kapasitas dalam permintaan `CreateFleet` 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.

Membuat templat peluncuran

ID templat peluncuran digunakan di langkah berikut. Untuk informasi selengkapnya, lihat [Membuat templat peluncuran](#).

Untuk mengonfigurasi templat peluncuran, pada `InstanceMarketOptionsRequest`, atur `MarketType` ke `capacity-block`. Tentukan ID reservasi Blok Kapasitas yang Anda targetkan dengan mengatur parameter `CapacityReservationID`.

Membuat Armada EC2

Buat 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": {
        "LaunchTemplateName": "CBR-launch-template",
        "Version": "1"
      },
      "Overrides": [
        {
          "InstanceType": "p5.48xlarge",
          "AvailabilityZone": "us-east-1a"
        }
      ]
    }
  ],
  "TargetCapacitySpecification": {
    "TotalTargetCapacity": 10,
    "DefaultTargetCapacityType": "capacity-block"
  },
  "Type": "instant"
}
```

Gunakan perintah [create-fleet](#) berikut.

```
aws ec2 create-fleet \
  --cli-input-json file://config.json
```

Untuk informasi selengkapnya, lihat [Membuat Armada EC2](#).

Tutorial: Menggunakan Armada Spot dengan pembobotan instans

Tutorial ini menggunakan perusahaan fiktif bernama Example Corp untuk mengilustrasikan proses permintaan Armada Spot menggunakan pembobotan instans.

Tujuan

Example Corp, sebuah perusahaan farmasi, ingin memanfaatkan 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 berikut untuk Armada Spot mereka.

Jenis instans

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 pembobotan instance, kapasitas target dapat sama dengan sejumlah instance (default) atau kombinasi faktor seperti inti (vCPU), memori (), dan penyimpanan (GBGiBs). Dengan mempertimbangkan dasar untuk aplikasi mereka (60 GB RAM dan delapan vCPU) sebagai 1 unit, Example Corp memutuskan bahwa 20 kali jumlah ini akan memenuhi kebutuhan mereka. Jadi, perusahaan menetapkan kapasitas target permintaan Armada Spot mereka menjadi 20.

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 Spot.

Harga per unit jam

Example Corp menggunakan [harga Sesuai Permintaan](#) per jam instans sebagai titik awal untuk harga mereka. 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:

Jenis instans	Harga Sesuai Permintaan	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 permintaan Armada Spot, Example Corp memverifikasi bahwa mereka memiliki peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya, lihat [Izin Armada Spot](#).

Membuat permintaan

Example Corp membuat file, `config.json`, dengan konfigurasi sebagai berikut untuk permintaan Armada Spot file tersebut:

```
{
  "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-482e4972",
    "WeightedCapacity": 1
  },
  {
    "ImageId": "ami-1a2b3c4d",
    "InstanceType": "r3.4xlarge",
    "SubnetId": "subnet-482e4972",
    "WeightedCapacity": 2
  },
  {
    "ImageId": "ami-1a2b3c4d",
    "InstanceType": "r3.8xlarge",
    "SubnetId": "subnet-482e4972",
    "SpotPrice": "0.90",
    "WeightedCapacity": 4
  }
]
}
```

Example Corp membuat permintaan Armada Spot menggunakan perintah [request-spot-fleet](#).

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Untuk informasi selengkapnya, lihat [Tipe permintaan Armada Spot](#).

Pemenuhan

Strategi alokasi menentukan asal dari kolam kapasitas Spot yang menjadi sumber Instans Spot Anda.

Dengan strategi `lowestPrice` (yang merupakan strategi default), Instans Spot berasal dari kolam dengan harga terendah per unit pada saat pemenuhan. Untuk menyediakan kapasitas 20 unit, Armada Spot 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 Spot 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.

Contoh konfigurasi Armada EC2 dan Armada Spot

Contoh berikut menunjukkan konfigurasi peluncuran yang dapat Anda gunakan untuk membuat Armada EC2 dan Armada Spot.

Topik

- [Contoh konfigurasi Armada EC2](#)
- [Konfigurasi contoh Armada Spot](#)

Contoh konfigurasi Armada EC2

Contoh berikut menunjukkan konfigurasi peluncuran yang dapat Anda gunakan dengan perintah [create-fleet](#) untuk membuat Armada EC2. Untuk informasi tentang parameter, lihat [create-fleet](#) di Referensi Perintah AWS CLI .

Contoh

- [Contoh 1: Meluncurkan Instans Spot sebagai opsi pembelian default](#)
- [Contoh 2: Meluncurkan Instans Sesuai Permintaan sebagai opsi pembelian default](#)
- [Contoh 3: Meluncurkan Instans Sesuai Permintaan sebagai kapasitas primer](#)
- [Contoh 4: Luncurkan Instans Sesuai Permintaan menggunakan beberapa Reservasi Kapasitas](#)
- [Contoh 5: Luncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak digunakan](#)
- [Contoh 6: Luncurkan Instans Sesuai Permintaan 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 di armada price-capacity-optimized](#)
- [Contoh 11: Konfigurasikan pemilihan tipe instans berbasis atribut](#)

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: Meluncurkan Instans Sesuai Permintaan 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 on-demand, yang menghasilkan armada meluncurkan 2 Instans Sesuai Permintaan.

```
{
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
      }
    }
  ],
  "TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "on-demand"
  }
}
```

```

    "TargetCapacitySpecification": {
      "TotalTargetCapacity": 2,
      "DefaultTargetCapacityType": "on-demand"
    }
  }
}

```

Contoh 3: Meluncurkan Instans Sesuai Permintaan sebagai kapasitas primer

Contoh berikut menentukan total kapasitas target dari 2 instans untuk armada tersebut dan kapasitas target dari 1 Instans Sesuai Permintaan. Opsi pembelian default adalah spot. Armada meluncurkan 1 Instans Sesuai Permintaan sebagaimana ditentukan, tetapi perlu meluncurkan satu instans lagi untuk memenuhi total kapasitas target. Opsi pembelian untuk selisihnya dihitung sebagai $\text{TotalTargetCapacity} - \text{OnDemandTargetCapacity} = \text{DefaultTargetCapacityType}$, 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 Instans Sesuai Permintaan menggunakan beberapa Reservasi Kapasitas

Anda dapat mengonfigurasi armada agar menggunakan Reservasi Kapasitas Sesuai Permintaan terlebih dahulu saat meluncurkan Instans Sesuai Permintaan dengan mengatur 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 Instans Sesuai Permintaan
- Total Reservasi Kapasitas yang tidak terpakai: 15 (lebih dari kapasitas target armada sebesar 12 Instans Sesuai Permintaan)
- Jumlah kolom Reservasi Kapasitas: 3 (m5.large, m4.xlarge, dan m4.2xlarge)
- Jumlah Reservasi Kapasitas per kolom: 5
- Strategi alokasi Sesuai Permintaan: lowest-price (Jika terdapat lebih dari satu Reservasi Kapasitas yang tidak terpakai di lebih dari satu kolom instans, armada akan menentukan kolom tempat untuk meluncurkan Instans Sesuai Permintaan berdasarkan strategi alokasi Sesuai Permintaan.)

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 kolom yang berbeda. Jumlah Reservasi Kapasitas di setiap kolom 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. Strategi alokasi Sesuai Permintaan adalah lowest-price. Strategi penggunaan untuk Reservasi Kapasitas adalah use-capacity-reservations-first.

Dalam contoh ini, harga Instans Sesuai Permintaan 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 Instans Sesuai Permintaan m5.large di us-east-1a – m5.large di us-east-1a merupakan harga terendah, dan terdapat 5 Reservasi Kapasitas m5.large yang tidak terpakai yang tersedia
- 5 Instans Sesuai Permintaan m4.xlarge di us-east-1a – m4.xlarge di us-east-1a merupakan harga terendah berikutnya, dan terdapat 5 Reservasi Kapasitas m4.xlarge yang tidak terpakai yang tersedia
- 2 Instans Sesuai Permintaan m4.2xlarge di us-east-1a – m4.2xlarge di us-east-1a merupakan harga terendah ketiga, dan terdapat 5 Reservasi Kapasitas m4.2xlarge yang hanya dibutuhkan 2 untuk memenuhi target kapasitas

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: Luncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak digunakan

Anda dapat mengonfigurasi armada agar menggunakan Reservasi Kapasitas Sesuai Permintaan terlebih dahulu saat meluncurkan Instans Sesuai Permintaan dengan mengatur strategi penggunaan untuk Reservasi Kapasitas ke `use-capacity-reservations-first`. Contoh ini juga menunjukkan cara armada memilih kolam instans tempat untuk meluncurkan Instans Sesuai Permintaan jika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia.

Dalam contoh ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 16 Instans Sesuai Permintaan
- Total Reservasi Kapasitas yang tidak terpakai: 15 (kurang dari kapasitas target armada sebesar 16 Instans Sesuai Permintaan)
- Jumlah kolam Reservasi Kapasitas: 3 (`m5.large`, `m4.xlarge`, dan `m4.2xlarge`)
- Jumlah Reservasi Kapasitas per kolam: 5

- Strategi alokasi Sesuai Permintaan: lowest-price (Jika jumlah Reservasi Kapasitas yang tidak terpakai kurang dari kapasitas target Sesuai Permintaan, armada akan menentukan kolam tempat meluncurkan kapasitas Sesuai Permintaan yang tersisa berdasarkan strategi alokasi Sesuai Permintaan.)

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. Strategi alokasi Sesuai Permintaan adalah `lowest-price`. Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

Dalam contoh ini, harga Instans Sesuai Permintaan 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 Instans Sesuai Permintaan m5.large di us-east-1a – m5.large di us-east-1a merupakan harga terendah, dan terdapat 5 Reservasi Kapasitas m5.large yang tidak terpakai yang tersedia. Reservasi Kapasitas digunakan terlebih dahulu untuk meluncurkan 5 Instans Sesuai Permintaan. Setelah sisa Reservasi Kapasitas m4.xlarge dan m4.2xlarge digunakan, untuk memenuhi kapasitas target, Instans Sesuai Permintaan tambahan diluncurkan sesuai dengan strategi alokasi Sesuai Permintaan, yaitu lowest-price dalam contoh ini.
- 5 Instans Sesuai Permintaan m4.xlarge di us-east-1a – m4.xlarge di us-east-1a merupakan harga terendah berikutnya, dan terdapat 5 Reservasi Kapasitas m4.xlarge yang tidak terpakai yang tersedia
- 5 Instans Sesuai Permintaan m4.2xlarge di us-east-1a – m4.2xlarge di us-east-1a merupakan harga terendah ketiga, dan terdapat 5 Reservasi Kapasitas m4.2xlarge 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: Luncurkan Instans Sesuai Permintaan menggunakan Reservasi Kapasitas yang ditargetkan

Anda dapat mengonfigurasi armada agar menggunakan Reservasi Kapasitas Sesuai Permintaan targeted terlebih dahulu saat meluncurkan Instans Sesuai Permintaan dengan mengatur strategi penggunaan untuk Reservasi Kapasitas ke `use-capacity-reservations-first`. Contoh ini menunjukkan cara meluncurkan Instans Sesuai Permintaan ke dalam Reservasi Kapasitas targeted, jika atribut Reservasi Kapasitas sama kecuali untuk Zona Ketersediaan (`us-east-1a` dan `us-east-1b`). Contoh ini juga menunjukkan cara armada memilih kolam instans tempat untuk meluncurkan Instans Sesuai Permintaan jika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia.

Dalam contoh ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 10 Instans Sesuai Permintaan
- Total Reservasi Kapasitas targeted yang tidak terpakai: 6 (kurang dari kapasitas target Sesuai Permintaan armada sebesar 10 Instans Sesuai Permintaan)
- Jumlah kolam Reservasi Kapasitas: 2 (`us-east-1a` dan `us-east-1b`)
- Jumlah Reservasi Kapasitas per kolam: 3

- Strategi alokasi Sesuai Permintaan: lowest-price (Jika jumlah Reservasi Kapasitas yang tidak terpakai kurang dari kapasitas target Sesuai Permintaan, armada akan menentukan kolam tempat meluncurkan kapasitas Sesuai Permintaan yang tersisa berdasarkan strategi alokasi Sesuai Permintaan.)

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: Meluncurkan Instans Sesuai Permintaan 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. Strategi alokasi Sesuai

Permintaan adalah `lowest-price`. Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

Dalam contoh ini, harga Instans Sesuai Permintaan untuk `c5.xlarge` di `us-east-1` adalah 0,17 USD 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 Instans Sesuai Permintaan sebagai berikut:
 - 3 Instans Sesuai Permintaan diluncurkan ke dalam 3 Reservasi Kapasitas `c5.xlarge` targeted di `us-east-1a`
 - 3 Instans Sesuai Permintaan diluncurkan ke dalam 3 Reservasi Kapasitas `c5.xlarge` targeted di `us-east-1b`
- Untuk memenuhi kapasitas target, 4 Instans Sesuai Permintaan tambahan diluncurkan ke kapasitas Sesuai Permintaan reguler sesuai dengan strategi alokasi Sesuai Permintaan, yaitu `lowest-price` dalam contoh ini. Namun, karena kolam memiliki harga yang sama (karena harganya adalah per Wilayah dan bukan per Zona Ketersediaan), armada meluncurkan 4 Instans Sesuai Permintaan yang tersisa ke salah satu kolam.

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: Konfigurasi 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 [Ops 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 kolom kapasitas Spot yang ditentukan dalam permintaan Armada EC2. Sebaiknya konfigurasi 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 [Penyeimbangan Ulang Kapasitas](#).

```
{
  "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 kolom kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada EC2 berupaya meluncurkan 50 Instans Spot ke kolom 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 di armada price-capacity-optimized

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 kolom kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada EC2 berupaya meluncurkan 50 Instans Spot ke kolom kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan sekaligus memilih kolom 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: Konfigurasi pemilihan tipe 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 kemungkinan atribut yang dapat Anda tentukan, lihat [InstanceRequirements](#) di Referensi 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"  
}
```

Konfigurasi contoh Armada Spot

Contoh berikut menunjukkan konfigurasi peluncuran yang dapat Anda gunakan dengan perintah [request-spot-fleet](#) untuk membuat permintaan Armada Spot. Untuk informasi selengkapnya, lihat [Membuat permintaan 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: Meluncurkan Instans Spot menggunakan Zona Ketersediaan atau subnet dengan harga terendah di Wilayah](#)
- [Contoh 2: Meluncurkan Instans Spot menggunakan Zona Ketersediaan atau subnet dengan harga terendah di daftar yang ditentukan](#)
- [Contoh 3: Meluncurkan Instans Spot menggunakan tipe instans dengan harga terendah dalam daftar yang ditentukan](#)
- [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: Meluncurkan Armada Spot dengan kapasitas Sesuai Permintaan](#)
- [Contoh 8: Mengonfigurasi 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 armada priceCapacityOptimized](#)

- [Contoh 12: Mengonfigurasi pemilihan tipe instans berbasis atribut](#)

Contoh 1: Meluncurkan Instans Spot menggunakan Zona Ketersediaan atau subnet 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 bayarkan tidak melebihi harga Sesuai Permintaan.

```
{
  "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: Meluncurkan Instans Spot menggunakan Zona Ketersediaan atau subnet dengan harga terendah di daftar yang ditentukan

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::880185128111:instance-profile/my-iam-role"
  }
}
...

```

Contoh 3: Meluncurkan Instans Spot menggunakan tipe instans dengan harga terendah dalam daftar yang ditentukan

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

Sebaiknya gunakan harga maksimum default, yaitu harga Sesuai Permintaan. 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 [Strategi alokasi untuk Instans Spot](#).

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 [Pembobotan instans Armada Spot](#).

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: Meluncurkan Armada Spot dengan kapasitas Sesuai Permintaan

Untuk memastikan bahwa Anda selalu memiliki kapasitas instans, Anda dapat menyertakan permintaan kapasitas Sesuai Permintaan dalam permintaan Armada Spot. Jika terdapat kapasitas, permintaan Sesuai Permintaan akan selalu terpenuhi. Keseimbangan kapasitas target akan terpenuhi sebagai Spot jika terdapat kapasitas dan ketersediaan.

Contoh berikut menentukan kapasitas target yang diinginkan sebagai 10, yang 5 di antaranya harus merupakan kapasitas Sesuai Permintaan. Kapasitas spot tidak ditentukan; hal tersebut tersirat dalam keseimbangan kapasitas target dikurangi kapasitas Sesuai Permintaan. Amazon EC2 meluncurkan 5

unit kapasitas sebagai Sesuai Permintaan, dan 5 unit kapasitas ($10 - 5 = 5$) sebagai Spot jika tersedia kapasitas dan ketersediaan Amazon EC2.

Untuk informasi selengkapnya, lihat [Sesuai Permintaan di Armada Spot](#).

```
{
  "IamFleetRole": "arn:aws:iam::781603563322: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: Mengonfigurasi 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 konfigurasi 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 [Penyeimbangan Ulang Kapasitas](#).

```
{
  "SpotFleetRequestConfig": {
    "AllocationStrategy": "capacityOptimized",
    "IamFleetRole": "arn:aws:iam::000000000000: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 kolom kapasitas Spot, yaitu makin rendah angkanya, makin tinggi prioritasnya. Anda juga dapat mengatur prioritas yang sama untuk beberapa kolom kapasitas Spot jika Anda menginginkannya setara. Jika Anda tidak menetapkan prioritas, kolom akan dianggap yang terakhir dalam hal prioritas.

Untuk memprioritaskan kolom 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 kolom kapasitas Spot. Setiap kolom diprioritaskan, yaitu makin rendah jumlahnya, makin tinggi prioritasnya. Kapasitas target adalah 50 Instans Spot. Armada Spot berupaya meluncurkan 50 Instans Spot ke dalam kolom 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 armada priceCapacityOptimized

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 kolom kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada Spot berupaya meluncurkan 50 Instans Spot ke kolom

kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan sekaligus memilih kolom 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 kemungkinan atribut yang dapat Anda tentukan, lihat [InstanceRequirements](#) di Referensi 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 armada

Kuota Amazon EC2 biasa (sebelumnya disebut sebagai batas) berlaku untuk instans yang diluncurkan oleh Armada EC2 atau Armada Spot, seperti [batas Instans Spot](#) dan [batas volume](#).

Selain itu, kuota berikut ini berlaku:

Deskripsi kuota	Kuota
Jumlah Armada EC2 dan Armada Spot per Wilayah jenis <code>maintain</code> dan <code>request</code> <code>diactive</code> , <code>deleted_running</code> dan negara bagian <code>cancelled_running</code>	1,000 ^{1 2 3}
Jumlah Armada EC2 tipe <code>instant</code>	Tidak terbatas.
Jumlah kumpulan kapasitas Spot (kombinasi unik dari tipe instans dan subnet) untuk Armada EC2 dan Armada Spot jenis dan <code>maintain request</code>	300 ¹
Jumlah kumpulan kapasitas Spot (kombinasi unik dari tipe instans dan subnet) untuk jenis Armada EC2 <code>instant</code>	Tidak terbatas.
Ukuran data pengguna dalam spesifikasi peluncuran	16 KB ²
Kapasitas target per Armada EC2 atau Armada Spot	10.000
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 peningkatan 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 [Hapus Armada EC2](#) dan [Membatalkan permintaan Armada Spot](#).

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 AWS Support Center [Create case](#).
2. Pilih Peningkatan batas layanan.
3. Untuk Tipe batas, pilih Armada EC2.
4. Untuk Wilayah, pilih AWS Wilayah tempat permintaan 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.

Memantau Amazon EC2

Pemantauan merupakan bagian penting dalam menjaga keandalan, ketersediaan, dan kinerja instans Amazon Elastic Compute Cloud (Amazon EC2) dan solusi Anda. AWS Anda harus mengumpulkan data pemantauan dari semua bagian dalam AWS solusi Anda sehingga Anda dapat lebih mudah men-debug kegagalan multi-titik jika terjadi. Namun, sebelum Anda mulai memantau Amazon EC2, Anda harus membuat rencana pemantauan yang harus mencakup:

- Apa saja tujuan pemantauan Anda?
- Apa saja sumber daya yang akan Anda pantau?
- Seberapa sering Anda akan memantau sumber daya ini?
- Alat pemantauan apa yang akan Anda gunakan?
- Siapa yang akan melakukan tugas pemantauan?
- Siapa yang harus diberi tahu saat terjadi kesalahan?

Setelah Anda menetapkan tujuan pemantauan dan membuat rencana pemantauan, langkah berikutnya adalah menetapkan batas dasar untuk performa Amazon EC2 normal di lingkungan Anda. Anda harus mengukur performa Amazon EC2 pada berbagai waktu dan di bawah kondisi beban yang berbeda. Saat memantau Amazon EC2, Anda harus menyimpan riwayat data pemantauan yang Anda kumpulkan. Anda dapat membandingkan performa Amazon EC2 saat ini dengan data historis ini untuk membantu Anda mengidentifikasi pola performa normal dan anomali performa, serta merancang metode untuk menanganinya. Misalnya, Anda dapat memantau pemanfaatan CPU, I/O disk, dan pemanfaatan jaringan untuk instans EC2 Anda. Ketika performa berada di luar batas dasar yang telah ditetapkan, Anda mungkin perlu mengonfigurasi ulang atau mengoptimalkan instans untuk mengurangi pemanfaatan CPU, meningkatkan I/O disk, atau mengurangi lalu lintas jaringan.

Untuk menetapkan batas dasar, setidaknya Anda harus memantau item berikut:

Item yang harus dipantau	Metrik Amazon EC2	Agen Pemantau/Log CloudWatch
Pemanfaatan CPU	CPUUtilization	
Pemanfaatan jaringan	NetworkIn NetworkOut	

Item yang harus dipantau	Metrik Amazon EC2	Agen Pemantau/Log CloudWatch
Kinerja disk	DiskReadOps DiskWriteOps	
Baca/Tulis Disk	DiskReadByte DiskWriteByte	
Pemanfaatan memori, pemanfaatan swap disk, pemanfaatan ruang disk, pemanfaatan file halaman, pengumpulan log		[Instans Linux dan Windows Server] Kumpulkan Metrik dan Log dari Instans Amazon EC2 dan Server Lokal dengan Agen CloudWatch [Migrasi dari agen CloudWatch Log sebelumnya pada instance Windows Server] Migrasikan Koleksi Log Instance Windows Server ke Agen CloudWatch

Pemantauan otomatis dan manual

AWS menyediakan berbagai alat yang dapat Anda gunakan untuk memantau Amazon EC2. Anda dapat mengonfigurasi beberapa alat tersebut guna melakukan pemantauan untuk Anda, sedangkan beberapa alat lainnya memerlukan intervensi manual.

Alat pemantauan

- [Alat pemantauan otomatis](#)
- [Alat pemantauan manual](#)

Alat pemantauan otomatis

Anda dapat menggunakan alat pemantauan otomatis berikut untuk memantau Amazon EC2 dan melaporkan kembali kepada Anda saat terjadi masalah:

- **Pemeriksaan status sistem** — memantau AWS sistem yang diperlukan untuk menggunakan instans Anda untuk memastikan bahwa mereka berfungsi dengan baik. Pemeriksaan ini mendeteksi masalah dengan instans Anda yang memerlukan AWS keterlibatan untuk memperbaiki. 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 Anda](#).

- **Pemeriksaan status instans** – memantau konfigurasi jaringan dan perangkat lunak pada tiap-tiap instans Anda secara terpisah. 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 Anda](#).

- **CloudWatch Alarm Amazon** — tonton satu metrik selama periode waktu yang Anda tentukan, dan lakukan satu atau beberapa 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 untuk perubahan status berkelanjutan saja. CloudWatch alarm tidak akan memanggil tindakan hanya karena mereka berada dalam keadaan tertentu; negara harus telah berubah dan dipertahankan untuk sejumlah periode tertentu. Untuk informasi selengkapnya, lihat [Pantau instans Anda menggunakan CloudWatch](#).

- Amazon EventBridge — mengotomatiskan AWS layanan Anda dan merespons secara otomatis peristiwa sistem. Acara dari AWS layanan dikirimkan ke EventBridge dalam waktu dekat, dan Anda dapat menentukan tindakan otomatis yang akan diambil saat acara cocok dengan aturan yang Anda tulis. Untuk informasi selengkapnya, lihat [Apa itu Amazon EventBridge?](#) .
- Amazon CloudWatch Logs — memantau, menyimpan, dan mengakses file log Anda dari instans Amazon EC2, AWS CloudTrail, atau sumber lain. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CloudWatch Logs](#).
- 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 Amazon EC2 dan Server Lokal dengan CloudWatch Agen di Panduan Pengguna Amazon CloudWatch](#)

Alat pemantauan manual

Bagian penting lainnya dari pemantauan Amazon EC2 melibatkan pemantauan secara manual item yang tidak dicakup oleh skrip pemantauan, pemeriksaan status, dan CloudWatch alarm. Dasbor Amazon EC2 dan CloudWatch konsol memberikan at-a-glance tampilan status lingkungan Amazon EC2 Anda.

- Dasbor Amazon EC2 menunjukkan:
 - Kondisi Layanan dan Peristiwa Terjadwal berdasarkan Wilayah
 - Status instans
 - Pemeriksaan status
 - Status alarm
 - Detail metrik instans (Di panel navigasi, pilih Instans, pilih satu instans, dan pilih tab Pemantauan)
 - Detail metrik volume (Di panel navigasi, pilih Volume, pilih satu volume, dan pilih tab Pemantauan)
- CloudWatch Dasbor Amazon menunjukkan:
 - Alarm dan status saat ini

- Grafik alarm dan sumber daya
- Status kesehatan layanan

Selain itu, Anda dapat menggunakan CloudWatch untuk melakukan hal berikut:

- Membuat grafik data pemantauan Amazon EC2 untuk memecahkan masalah dan menemukan tren
- Cari dan telusuri semua metrik AWS sumber daya Anda
- Membuat dan mengedit alarm untuk menerima notifikasi terkait masalah
- Lihat at-a-glance ikhtisar alarm dan sumber daya Anda AWS

Praktik terbaik untuk pemantauan

Gunakan praktik terbaik untuk pemantauan berikut agar dapat membantu Anda dalam tugas pemantauan Amazon EC2.

- Jadikan pemantauan sebagai prioritas untuk mengatasi masalah kecil sebelum menjadi masalah besar.
- Buat dan terapkan rencana pemantauan yang mengumpulkan data pemantauan dari semua bagian dalam AWS solusi Anda sehingga Anda dapat lebih mudah men-debug kegagalan multi-titik jika terjadi. Rencana pemantauan Anda setidaknya harus menjawab pertanyaan-pertanyaan berikut:
 - Apa saja tujuan pemantauan Anda?
 - Apa saja sumber daya yang akan Anda pantau?
 - Seberapa sering Anda akan memantau sumber daya ini?
 - Alat pemantauan apa yang akan Anda gunakan?
 - Siapa yang akan melakukan tugas pemantauan?
 - Siapa yang harus diberi tahu saat terjadi kesalahan?
- Otomatiskan tugas pemantauan sebanyak mungkin.
- Periksa file log instans EC2 Anda.

Memantau status instans 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 peristiwa 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 Anda](#)
- [Peristiwa perubahan status untuk instans Anda](#)
- [Peristiwa terjadwal untuk instans Anda](#)

Pemeriksaan status untuk instans Anda

Dengan pemantauan status instans, Anda dapat dengan cepat menentukan apakah Amazon EC2 telah mendeteksi masalah yang mungkin mencegah instans Anda dari menjalankan aplikasi. Amazon EC2 melakukan pemeriksaan otomatis pada setiap instans EC2 yang berjalan untuk mengidentifikasi masalah perangkat keras dan perangkat lunak. Anda dapat melihat hasil dari pemeriksaan status ini untuk mengidentifikasi masalah spesifik yang dapat dideteksi. Data status peristiwa menambah informasi yang telah disediakan Amazon EC2 tentang status setiap instance (pending,seperitirunning,stopping,) dan metrik pemanfaatan yang dipantau CloudWatch Amazon (pemanfaatan CPU, lalu lintas jaringan, dan aktivitas disk).

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 dibangun di Amazon EC2 sehingga pemeriksaan tidak dapat dinonaktifkan atau dihapus.

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 CloudWatch alarm 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 [Membuat dan mengedit alarm pemeriksaan status](#).

Anda juga dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2 dan memulihkan instans secara otomatis jika menjadi rusak karena masalah mendasar. Untuk informasi selengkapnya, lihat [Ketahanan contoh](#).

Daftar Isi

- [Tipe pemeriksaan status](#)
- [Bekerja dengan pemeriksaan status](#)

Tipe pemeriksaan status

Ada tiga jenis pemeriksaan status.

- [Pemeriksaan status sistem](#)
- [Pemeriksaan status instans](#)
- [Pemeriksaan status EBS terlampir](#)

Pemeriksaan status sistem

Pemeriksaan status sistem memantau AWS sistem tempat instans 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 Linux yang didukung oleh penyimpanan instans, Anda dapat mengakhiri dan mengganti instans tersebut. Untuk instans Windows, volume root harus berupa volume Amazon EBS. Penyimpanan instans tidak didukung untuk volume root. 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 memulai ulang dari sistem operasi pada instans bare metal, pemeriksaan status sistem tersebut mungkin kembali ke status gagal untuk sementara. Ketika instans tersedia, pemeriksaan status sistem seharusnya kembali ke status lulus.

Pemeriksaan status instans

Pemeriksaan status instans memantau konfigurasi jaringan dan perangkat lunak pada tiap-tiap instans Anda secara terpisah. 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 lainnya yang juga dapat mempengaruhi kesehatan instance. Pemeriksaan status yang bergantung pada ketersediaan jaringan dapat melaporkan status yang 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
- [Instans Windows] Selama reboot instance atau saat instance yang didukung toko instance Windows sedang dibundel, pemeriksaan status instance melaporkan kegagalan hingga instance tersedia lagi.

Jika pemeriksaan status instance gagal, kami menambah metrik [StatusCheckFailed_Instance](#).

Instans bare metal

Jika Anda memulai ulang dari sistem operasi pada instans bare metal, pemeriksaan status instans tersebut mungkin akan kembali ke status gagal untuk sementara. Ketika instans tersedia, pemeriksaan status instans seharusnya kembali ke status lulus.

Pemeriksaan status EBS terlampir

Anda dapat menggunakan pemeriksaan status EBS terlampir untuk memantau apakah volume Amazon EBS yang dilampirkan ke instans dapat dijangkau dan dapat menyelesaikan operasi I/O. Metrik `StatusCheckFailed_AttachedEBS` adalah nilai biner yang menunjukkan gangguan jika satu atau lebih volume EBS yang terlampir pada instans tidak dapat menyelesaikan operasi I/O. Pemeriksaan status ini mendeteksi masalah yang mendasari komputasi atau infrastruktur Amazon EBS. Jika metrik pemeriksaan status EBS terlampir 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 kinerja I/O dari setiap volume yang terpasang

menggunakan CloudWatch metrik EBS untuk mendeteksi dan mengganti volume yang terganggu. Jika beban kerja Anda tidak mendorong I/O ke salah satu volume EBS yang dilampirkan pada instans dan pemeriksaan status EBS terlampir menunjukkan adanya gangguan, Anda dapat menghentikan dan memulai instans untuk mengatasi masalah dengan host fisik yang memengaruhi jangkauan volume EBS. Untuk informasi selengkapnya, lihat [CloudWatch Metrik Amazon untuk Amazon EBS](#)

Note

- Metrik pemeriksaan status EBS yang terlampir hanya tersedia untuk instans Nitro.
- Anda dapat memantau metrik pemeriksaan status EBS terlampir dengan [membuat CloudWatch alarm](#) berdasarkan `StatusCheckFailed_AttachedEBS` metrik. Anda tidak dapat melihat pemeriksaan status ini dengan menggunakan [perintah AWS CLI `describe-instance-status`](#).

Bekerja dengan pemeriksaan status

Anda dapat menangani pemeriksaan status menggunakan konsol dan alat baris perintah, seperti AWS CLI.

Topik

- [Melihat pemeriksaan status](#)
- [Membuat dan mengedit alarm pemeriksaan status](#)

Melihat pemeriksaan status

Untuk melihat pemeriksaan status, gunakan salah satu metode berikut.

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.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availi
spot-instance-2	i-01aeed690c9fb5322	Running	t3.nano	1/2 checks ...	View alarms +	eu-w
spot-instance-1	i-0ba5e5bbc9d634fa6	Stopped	t3.nano	-	View alarms +	eu-w
EIC-RHEL	i-08e66e73da739c7f4	Running	t2.micro	2/2 checks passed	View alarms +	eu-w
Windows	i-0cb952751a0d8388b	Running	t3.nano	2/2 checks passed	View alarms +	eu-w

Instance: i-01aeed690c9fb5322 (spot-instance-2)

Details | **Status and alarms New** | Monitoring | Security | Networking | Storage | Tags

Status checks Info

Status checks detect problems that may impair i-01aeed690c9fb5322 (spot-instance-2) from running your applications.

System status checks

- System reachability check passed

► Metrics

▼ Alarms

Instance status checks

- Instance reachability check failed

Check failure at 2020/12/16 17:30 GMT+2 (about 1 month)

Find alarms by name

Name	State	Description	Metric name	State reason
Instance has no associated alarms				

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 masalah kegagalan pemeriksaan status sistem atau instans pada instance Linux, lihat. [Memecahkan masalah instance Linux dengan pemeriksaan status yang gagal](#)

- 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

Untuk informasi selengkapnya, lihat [the section called “Metrik pemeriksaan status”](#).

Command line

Anda dapat melihat pemeriksaan status untuk instans yang berjalan menggunakan perintah [describe-instance-status](#) (AWS CLI).

Untuk melihat status semua instans, gunakan perintah berikut.

```
aws ec2 describe-instance-status
```

Untuk mendapatkan status dari semua instans dengan status instans `impaired`, gunakan perintah berikut.

```
aws ec2 describe-instance-status \
  --filters Name=instance-status.status,Values=impaired
```

Untuk mendapatkan status instans tunggal, gunakan perintah berikut.

```
aws ec2 describe-instance-status \
  --instance-ids i-1234567890abcdef0
```

Atau, gunakan perintah berikut:

- [Get-EC2InstanceStatus](#) (AWS Tools for Windows PowerShell)
- [DescribeInstanceStatus](#) (API Kueri Amazon EC2)

Jika Anda memiliki instance Linux dengan pemeriksaan status gagal, lihat [Memecahkan masalah instance Linux dengan pemeriksaan status yang gagal](#).

Membuat dan mengedit alarm pemeriksaan status

Anda dapat menggunakan [metrik pemeriksaan status](#) untuk membuat CloudWatch alarm untuk memberi tahu Anda ketika sebuah instans memiliki pemeriksaan status yang gagal.

Important

Pemeriksaan status dan alarm pemeriksaan status untuk sementara dapat memasukkan status data yang tidak mencukupi jika ada titik data metrik yang hilang. Meskipun jarang, ini bisa terjadi ketika ada gangguan dalam sistem pelaporan metrik, bahkan ketika sebuah instance sehat. Sebaiknya Anda memperlakukan status ini sebagai data yang hilang, bukan kegagalan pemeriksaan status atau pelanggaran alarm, terutama saat menghentikan, menghentikan, mem-boot ulang, atau memulihkan tindakan pada instance sebagai tanggapan.

Untuk melihat pemeriksaan status, gunakan salah satu metode berikut:

Console

Gunakan prosedur berikut untuk mengonfigurasi alarm yang mengirim Anda notifikasi melalui email, atau menghentikan, mengakhiri, atau memulihkan instans saat gagal dalam pemeriksaan status.

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 akan mengirimkan pesan email konfirmasi langganan ke setiap alamat baru. Setiap penerima harus mengonfirmasi langganan dengan memilih tautan yang terdapat dalam pesan tersebut. Notifikasi pemberitahuan dikirim hanya ke alamat yang dikonfirmasi.

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 membuat perubahan pada alarm status instans, 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 instance dan pilih Actions, Monitoring, Manage 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.

Command line

Dalam contoh berikut, alarm menerbitkan notifikasi ke topik SNS, `arn:aws:sns:us-west-2:111122223333:my-sns-topic`, saat instans gagal dalam pemeriksaan instans ataupun pemeriksaan status sistem setidaknya untuk dua periode berturut-turut. CloudWatch Metrik yang digunakan adalah `StatusCheckFailed`.

Untuk membuat alarm pemeriksaan status menggunakan AWS CLI

1. Pilih topik SNS yang ada atau buat baru. Untuk informasi selengkapnya, lihat [Menggunakan Amazon SNS](#) di AWS Command Line Interface Panduan Pengguna. AWS CLI
2. Gunakan perintah [list-metrics berikut untuk melihat metrik](#) Amazon yang tersedia untuk Amazon CloudWatch 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-action arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

```
--alarm-actions arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

Periode adalah kerangka waktu, dalam hitungan detik, di mana CloudWatch metrik Amazon 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. Contoh ini mengonfigurasi alarm untuk mengirim email menggunakan Amazon SNS.

Peristiwa perubahan status untuk instans Anda

Amazon EC2 mengirimkan EC2 Instance State-change Notification peristiwa 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-abcd1111"
  ],
  "detail": {
    "instance-id": "i-abcd1111",
    "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`.

Mendapatkan notifikasi email saat status instans berubah

Untuk menerima pemberitahuan email saat instans Anda mengubah status, buat topik Amazon SNS, lalu buat EventBridge aturan untuk acara tersebut `EC2 Instance State-change Notification`.

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 Tipe peristiwa, pilih Notifikasi State-change Instans 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.

Peristiwa terjadwal untuk instans Anda

AWS dapat menjadwalkan acara untuk instance Anda, seperti reboot, stop/start, atau pensiun. Peristiwa ini tidak sering terjadi. Jika salah satu instans Anda akan terpengaruh oleh acara yang dijadwalkan, AWS kirimkan email ke alamat email yang terkait dengan AWS akun Anda sebelum acara yang dijadwalkan. Email tersebut memberikan detail tentang peristiwa, termasuk tanggal mulai dan berakhir. Tergantung pada 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 CloudWatch Events. Untuk informasi selengkapnya tentang memantau AWS Health peristiwa dengan CloudWatch, lihat [Memantau AWS Health peristiwa dengan CloudWatch Acara](#).

Acara terjadwal dikelola oleh AWS; Anda tidak dapat menjadwalkan acara untuk instans Anda. Anda dapat melihat acara yang dijadwalkan oleh AWS, menyesuaikan pemberitahuan acara terjadwal

untuk menyertakan atau menghapus tag dari pemberitahuan email, dan melakukan tindakan saat instance dijadwalkan untuk reboot, pensiun, atau berhenti.

Untuk memperbarui informasi kontak akun agar Anda dapat memastikan akan diberi tahu tentang peristiwa terjadwal, buka halaman [Pengaturan 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 kondisi yang dilakukan oleh Amazon EC2 Auto Scaling, lihat [Pemeriksaan kondisi untuk instans Penskalaan Otomatis](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.

Daftar Isi

- [Tipe peristiwa terjadwal](#)
- [Melihat peristiwa terjadwal](#)
- [Mengustomisasi notifikasi peristiwa terjadwal](#)
- [Bekerja dengan instans yang dijadwalkan untuk berhenti atau pensiun](#)
- [Menjadwalkan boot ulang instans](#)
- [Menjadwalkan pemeliharaan instans](#)
- [Menjadwalkan ulang peristiwa terjadwal](#)
- [Menentukan jendela peristiwa untuk peristiwa terjadwal](#)

Tipe peristiwa terjadwal

Amazon EC2 dapat membuat tipe peristiwa berikut untuk instans Anda, tempat peristiwa tersebut terjadi pada waktu terjadwal:

- Penghentian instans: Pada waktu yang dijadwalkan, instans dihentikan. Saat Anda memulainya lagi, instans dimigrasikan ke host baru. Berlaku hanya untuk instans yang didukung oleh Amazon EBS.
- Pemensiunan instans: Pada waktu yang dijadwalkan, instans dihentikan jika didukung oleh Amazon EBS, atau diakhiri jika didukung oleh penyimpanan instans.

- Boot ulang instans: Pada waktu yang dijadwalkan, instans di-boot ulang.
- Boot ulang sistem: Pada waktu yang dijadwalkan, host untuk instans di-boot ulang.
- Pemeliharaan sistem: Pada waktu yang dijadwalkan, instans mungkin akan terpengaruh untuk sementara oleh pemeliharaan jaringan atau pemeliharaan daya.

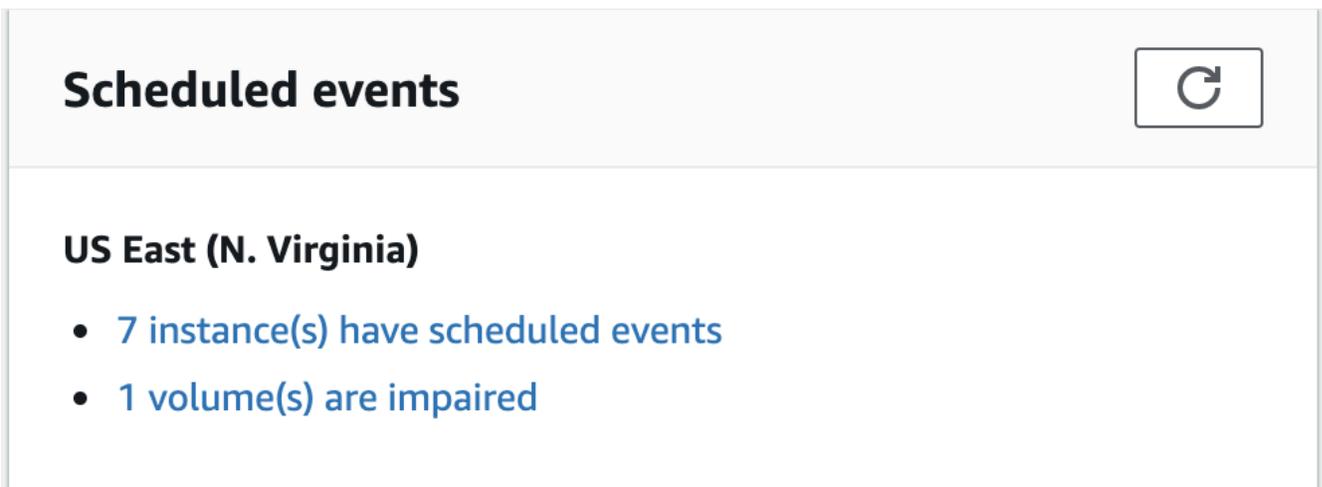
Melihat peristiwa terjadwal

Selain menerima notifikasi peristiwa terjadwal di email, Anda dapat memeriksa peristiwa terjadwal menggunakan salah satu metode berikut.

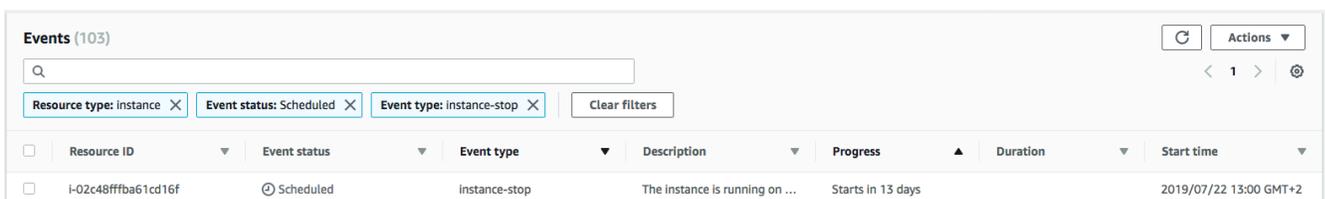
Console

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.



Resource ID	Event status	Event type	Description	Progress	Duration	Start time
I-02c48ffba61cd16f	Scheduled	Instance-stop	The instance is running on ...	Starts in 13 days		2019/07/22 13:00 GMT+2

AWS CLI

Untuk melihat peristiwa terjadwal instans Anda

Gunakan perintah [describe-instance-status](#).

```
aws ec2 describe-instance-status \  
  --instance-id i-1234567890abcdef0 \  
  --query "InstanceStatuses[0].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"  
    }  
  ]  
]
```

Contoh output berikut menunjukkan peristiwa pemensiunan 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 untuk instans Anda menggunakan AWS Tools for Windows PowerShell

Gunakan perintah berikut [Get-EC2InstanceStatus](#).

```
PS C:\> (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 AWS Health Dashboard untuk mempelajari tentang peristiwa yang dapat memengaruhi instans Anda. Ini AWS Health Dashboard mengatur 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 [Memulai dengan Anda AWS Health Dashboard](#) di Panduan AWS Health Pengguna.

Mengustomisasi notifikasi peristiwa terjadwal

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.

Anda dapat menyertakan tanda dalam notifikasi peristiwa menggunakan salah satu metode 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 AWS CLI [register-instance-event-notification-attributes](#) dan atur parameter `IncludeAllTagsOfInstance` ke `true`.

```
aws ec2 register-instance-event-notification-attributes \
```

```
--instance-tag-attribute "IncludeAllTagsOfInstance=true"
```

Untuk menyertakan tanda tertentu dalam notifikasi peristiwa

Gunakan perintah AWS CLI [register-instance-event-notification-attributes](#) dan tentukan tanda yang akan disertakan menggunakan parameter InstanceTagKeys.

```
aws ec2 register-instance-event-notification-attributes \  
  --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",  
  "tag_key_3"]'
```

Menghapus tanda dari notifikasi peristiwa

Anda dapat menghapus tanda dari notifikasi peristiwa menggunakan salah satu metode berikut.

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 AWS CLI [deregister-instance-event-notification-attributes](#) dan atur parameter IncludeAllTagsOfInstance ke false.

```
aws ec2 deregister-instance-event-notification-attributes \  
  --instance-tag-attribute "IncludeAllTagsOfInstance=false"
```

Untuk menghapus tanda tertentu dari notifikasi peristiwa

Gunakan AWS CLI perintah [deregister-instance-event-notification-attributes](#) dan tentukan tag yang akan dihapus dengan menggunakan InstanceTagKeys parameter.

```
aws ec2 deregister-instance-event-notification-attributes \  
  --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",  
  "tag_key_3"]'
```

Melihat tanda yang akan disertakan dalam notifikasi peristiwa

Anda dapat melihat tanda yang akan disertakan dalam notifikasi peristiwa menggunakan salah satu metode berikut.

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 tanda yang akan disertakan dalam notifikasi peristiwa

Gunakan AWS CLI perintah [describe-instance-event-notification-attributes](#).

```
aws ec2 describe-instance-event-notification-attributes
```

Bekerja dengan instans yang dijadwalkan untuk berhenti atau pensiun

Ketika AWS mendeteksi kegagalan yang tidak dapat diperbaiki dari host yang mendasari untuk instans Anda, itu menjadwalkan instance untuk berhenti atau dihentikan, tergantung pada jenis perangkat root untuk instance tersebut. Jika perangkat root adalah volume EBS, instans dijadwalkan untuk dihentikan. Jika perangkat root adalah volume penyimpanan instans, instans dijadwalkan untuk diakhiri. Untuk informasi selengkapnya, lihat [Pensiun instans](#).

Important

Semua data yang disimpan pada volume penyimpanan instans hilang saat instans dihentikan, dihibernasi, atau diakhiri. Termasuk di dalamnya volume penyimpanan instans yang dilampirkan ke instans yang memiliki volume EBS sebagai perangkat root. Pastikan untuk menyimpan data dari volume penyimpanan instans yang mungkin Anda perlukan nanti sebelum instans dihentikan, dihibernasi, atau diakhiri.

Tindakan untuk Instans yang Didukung oleh Amazon EBS

Anda dapat menunggu instans untuk berhenti sesuai jadwal. Atau, Anda dapat menghentikan dan memulai sendiri instans, sehingga instans tersebut dimigrasikan ke host baru. Untuk informasi selengkapnya tentang penghentian instans, selain informasi tentang perubahan pada konfigurasi instans Anda saat dihentikan, lihat [Hentikan dan mulai instans Amazon EC2](#).

Anda dapat mengotomatiskan penghentian dan pemulaian langsung sebagai respons atas peristiwa penghentian instans terjadwal. Untuk informasi selengkapnya, lihat [Mengotomatiskan tindakan untuk instans Amazon EC2](#) di Panduan Pengguna AWS Health .

Tindakan untuk Instans yang Didukung oleh Penyimpanan Instans

Kami menyarankan Anda untuk meluncurkan instans pengganti dari AMI terbaru Anda dan memigrasikan semua data yang diperlukan ke instans pengganti tersebut sebelum instans dijadwalkan untuk diakhiri. Selanjutnya, Anda dapat mengakhiri instans asli atau menunggu hingga instans tersebut berakhir sesuai jadwal.

Menjadwalkan boot ulang instans

Ketika AWS harus melakukan tugas-tugas seperti menginstal pembaruan atau memelihara host yang mendasarinya, itu dapat menjadwalkan instance atau host yang mendasarinya untuk reboot. Anda dapat [menjadwalkan kembali sebagian besar peristiwa boot ulang](#) sehingga instans di-boot ulang pada tanggal dan waktu tertentu yang sesuai untuk Anda.

Melihat tipe peristiwa boot ulang

Anda dapat melihat apakah peristiwa boot ulang adalah boot ulang instans atau boot ulang sistem menggunakan salah satu metode berikut.

Console

Untuk melihat tipe peristiwa boot ulang terjadwal

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. Untuk setiap instans, lihat nilai pada kolom Tipe peristiwa. Nilainya adalah `system-reboot` atau `instance-reboot`.

AWS CLI

Untuk melihat tipe peristiwa boot ulang terjadwal

Gunakan perintah [describe-instance-status](#).

```
aws ec2 describe-instance-status \  
  --instance-id i-1234567890abcdef0
```

Untuk peristiwa boot ulang terjadwal, nilai untuk Code adalah `system-reboot` atau `instance-reboot`. Contoh output berikut menunjukkan peristiwa `system-reboot`.

```
[  
  "Events": [  
    {  
      "InstanceEventId": "instance-event-0d59937288b749b32",  
      "Code": "system-reboot",  
      "Description": "The instance is scheduled for a reboot",  
      "NotAfter": "2019-03-14T22:00:00.000Z",  
      "NotBefore": "2019-03-14T20:00:00.000Z",  
      "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"  
    }  
  ]  
]
```

Tindakan untuk boot ulang instans

Anda dapat menunggu hingga boot ulang instans dilakukan dalam jendela pemeliharaan terjadwal, [menjadwalkan kembali](#) boot ulang instans ke tanggal dan waktu yang sesuai untuk Anda, atau [melakukan boot ulang](#) instans sendiri pada waktu yang Anda inginkan.

Setelah instans Anda di-boot ulang, peristiwa terjadwal akan dihapus dan deskripsi peristiwa diperbarui. Pemeliharaan yang tertunda untuk host yang mendasari telah selesai, dan Anda dapat mulai menggunakan kembali instans setelah di-boot sepenuhnya.

Tindakan untuk boot ulang sistem

Anda tidak dapat melakukan sendiri boot ulang sistem. Anda dapat menunggu hingga boot ulang sistem dilakukan selama jendela pemeliharaan terjadwal atau [menjadwalkan kembali](#) boot ulang sistem ke tanggal dan waktu yang sesuai untuk Anda. Boot ulang sistem biasanya selesai dalam hitungan menit. Setelah boot ulang sistem dilakukan, instans akan mempertahankan alamat IP dan nama DNS miliknya. Semua data pada volume penyimpanan instans lokal juga dipertahankan. Setelah boot ulang sistem selesai, peristiwa yang dijadwalkan untuk instans tersebut akan dihapus, dan Anda dapat memverifikasi bahwa perangkat lunak pada instans Anda beroperasi seperti yang diharapkan.

Atau, jika perlu untuk mempertahankan instans pada waktu yang berbeda dan Anda tidak dapat menjadwalkan kembali boot ulang sistem, Anda dapat menghentikan dan memulai instans yang didukung oleh Amazon EBS, sehingga instans tersebut akan dimigrasikan ke host baru. Namun, data pada volume penyimpanan instans lokal tidak disimpan. Anda juga dapat mengotomatisasi penghentian dan pemulaian instans langsung sebagai respons atas peristiwa boot ulang sistem terjadwal. Untuk informasi selengkapnya, lihat [Mengotomatisasi Tindakan untuk Instans EC2](#) dalam Panduan Pengguna AWS Health . Untuk instans berbasis penyimpanan instans, jika Anda tidak dapat menjadwalkan ulang boot ulang sistem, Anda dapat meluncurkan instans pengganti dari AMI terbaru, memigrasikan semua data yang diperlukan ke instans pengganti sebelum jendela pemeliharaan terjadwal, lalu mengakhiri instans asli.

Menjadwalkan pemeliharaan instans

Ketika AWS harus memelihara 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 untuk Instans yang Didukung oleh Amazon EBS

Anda dapat menunggu hingga pemeliharaan dilakukan sesuai jadwal. Atau, Anda dapat menghentikan dan memulai instans, sehingga instans tersebut dimigrasikan ke host baru. Untuk informasi selengkapnya tentang penghentian instans, selain informasi tentang perubahan pada konfigurasi instans Anda saat dihentikan, lihat [Hentikan dan mulai instans Amazon EC2](#).

Anda dapat mengotomatisasi penghentian dan pemulaian langsung sebagai respons atas peristiwa pemeliharaan terjadwal. Untuk informasi selengkapnya, lihat [Mengotomatisasi Tindakan untuk Instans EC2](#) dalam Panduan Pengguna AWS Health .

Tindakan untuk Instans yang Didukung oleh Penyimpanan Instans

Anda dapat menunggu hingga pemeliharaan dilakukan sesuai jadwal. Atau, jika ingin mempertahankan operasi normal selama jendela pemeliharaan terjadwal, Anda dapat meluncurkan instans pengganti dari AMI terbaru, memigrasikan semua data yang diperlukan ke instans pengganti sebelum jendela pemeliharaan terjadwal, lalu mengakhiri instans asli.

Menjadwalkan ulang peristiwa terjadwal

Anda dapat menjadwalkan ulang peristiwa agar terjadi pada tanggal dan waktu tertentu yang sesuai untuk Anda. Hanya peristiwa dengan batas waktu yang dapat dijadwalkan ulang. Ada [batasan lain untuk menjadwalkan ulang peristiwa](#).

Anda dapat menjadwalkan ulang peristiwa menggunakan salah satu metode berikut.

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

1. Hanya peristiwa dengan tanggal batas waktu, yang ditunjukkan dengan nilai untuk `NotBeforeDeadline`, yang dapat dijadwalkan ulang. Gunakan perintah [describe-instance-status](#) untuk melihat nilai parameter `NotBeforeDeadline`.

```
aws ec2 describe-instance-status \  
  --instance-id i-1234567890abcdef0
```

Contoh output berikut menunjukkan peristiwa `system-reboot` yang dapat dijadwalkan ulang karena `NotBeforeDeadline` berisi suatu nilai.

```
[  
  "Events": [  
    {  
      "InstanceEventId": "instance-event-0d59937288b749b32",  
      "Code": "system-reboot",  
      "Description": "The instance is scheduled for a reboot",  
      "NotAfter": "2019-03-14T22:00:00.000Z",  
      "NotBefore": "2019-03-14T20:00:00.000Z",  
      "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"  
    }  
  ]  
]
```

]

2. Untuk menjadwalkan ulang peristiwa, gunakan perintah [modify-instance-event-start-time](#). Tentukan waktu mulai peristiwa baru menggunakan parameter `not-before`. Waktu mulai peristiwa baru harus jatuh sebelum `NotBeforeDeadline`.

```
aws ec2 modify-instance-event-start-time \  
  --instance-id i-1234567890abcdef0 \  
  --instance-event-id instance-event-0d59937288b749b32 \  
  --not-before 2019-03-25T10:00:00.000
```

Mungkin diperlukan waktu satu atau dua menit sebelum perintah [describe-instance-status](#) menampilkan nilai parameter `not-before` yang terbaru.

Batasan

- Hanya peristiwa dengan tanggal batas waktu yang dapat dijadwalkan ulang. Peristiwa dapat dijadwalkan ulang hingga tanggal batas waktu. Kolom `Batas waktu` di konsol dan `NotBeforeDeadline` bidang di AWS CLI menunjukkan jika acara memiliki tanggal tenggat waktu.
- Hanya peristiwa yang belum dimulai yang dapat dijadwalkan ulang. Kolom `Waktu mulai` di konsol dan `NotBefore` bidang di AWS CLI menunjukkan waktu mulai acara. Peristiwa yang dijadwalkan untuk dimulai dalam waktu 5 menit berikutnya 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.

Menentukan jendela peristiwa untuk peristiwa terjadwal

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 tidak terjadwal seperti AutoRecovery dan reboot yang tidak direncanakan.

Bekerja dengan jendela peristiwa

- [Pertimbangan](#)
- [Melihat jendela peristiwa](#)
- [Membuat jendela peristiwa](#)
- [Memodifikasi jendela peristiwa](#)
- [Menghapus jendela peristiwa](#)
- [Menandai jendela peristiwa](#)

Pertimbangan

- Format waktu semua jendela peristiwa adalah UTC.
- Durasi jendela peristiwa mingguan minimum adalah 4 jam.
- Rentang waktu dalam jendela peristiwa masing-masing setidaknya harus mencapai 2 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 AWS Wilayah.
- Banyak instans yang terkait dengan jendela peristiwa berpotensi memiliki peristiwa terjadwal yang terjadi pada saat bersamaan.
- Jika AWS telah 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 instans sebelum peristiwa terjadwal, sehingga instans tersebut akan dimigrasikan ke host baru, dan peristiwa terjadwal tidak akan lagi berlangsung.

Melihat jendela peristiwa

Anda dapat melihat jendela peristiwa menggunakan salah satu metode berikut.

Console

Untuk melihat 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 untuk melihat detailnya.

AWS CLI

Untuk mendeskripsikan semua jendela peristiwa

Gunakan perintah [describe-instance-event-windows](#).

```
aws ec2 describe-instance-event-windows \  
  --region us-east-1
```

Output yang diharapkan

```
{  
  "InstanceEventWindows": [  
    {
```

```

    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-0598c7d356eba48d7"
      ],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "active",
    "Tags": []
  }
  ...
],
"NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
}

```

Untuk mendeskripsikan jendela peristiwa tertentu

Gunakan perintah [describe-instance-event-windows](#) dengan parameter `--instance-event-window-id` untuk mendeskripsikan jendela peristiwa tertentu.

```

aws ec2 describe-instance-event-windows \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890

```

Untuk mendeskripsikan jendela peristiwa yang cocok dengan satu filter atau lebih

Gunakan perintah [describe-instance-event-windows](#) dengan parameter `--filters`. Dalam contoh berikut, filter `instance-id` digunakan untuk mendeskripsikan semua jendela peristiwa yang terkait dengan instans yang ditentukan.

Saat digunakan, filter melakukan pencocokan langsung. Namun, filter `instance-id` berbeda. Jika tidak ada kecocokan langsung dengan ID instans, filter akan menampilkan kembali asosiasi jendela peristiwa yang memiliki keterkaitan tidak langsung, seperti tanda instans atau ID Host Khusus (jika instans berada di Host Khusus).

Untuk daftar filter yang didukung, lihat [describe-instance-event-windows](#) dalam Referensi AWS CLI .

```
aws ec2 describe-instance-event-windows \  
  --region us-east-1 \  
  --filters Name=instance-id,Values=i-1234567890abcdef0 \  
  --max-results 100 \  
  --next-token <next-token-value>
```

Output yang diharapkan

Dalam contoh berikut, instans berada di Host Khusus yang terkait dengan jendela peristiwa.

```
{  
  "InstanceEventWindows": [  
    {  
      "InstanceEventWindowId": "iew-0dbc0adb66f235982",  
      "TimeRanges": [  
        {  
          "StartWeekDay": "sunday",  
          "StartHour": 2,  
          "EndWeekDay": "sunday",  
          "EndHour": 8  
        }  
      ],  
      "Name": "myEventWindowName",  
      "AssociationTarget": {  
        "InstanceIds": [],  
        "Tags": [],  
        "DedicatedHostIds": [  
          "h-0140d9a7ecbd102dd"  
        ]  
      },  
      "State": "active",  
      "Tags": []  
    }  
  ]  
}
```

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.

Untuk batasan jendela peristiwa, lihat [Pertimbangan](#) yang dibahas sebelumnya dalam topik ini.

Jendela peristiwa berulang setiap pekan sampai Anda menghapusnya.

Gunakan salah satu metode berikut untuk membuat jendela peristiwa.

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 beberapa instans dengan jendela peristiwa sehingga jika instans tersebut dijadwalkan untuk pemeliharaan, peristiwa terjadwal akan terjadi selama jendela peristiwa terkait. Anda dapat mengaitkan satu atau beberapa instans dengan jendela peristiwa menggunakan ID instans atau tanda instans. Anda dapat mengaitkan Host Khusus dengan jendela peristiwa menggunakan ID host.

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 acara menggunakan AWS CLI, Anda pertama kali membuat jendela acara, dan kemudian Anda mengaitkan satu atau beberapa target dengan jendela acara.

Membuat jendela peristiwa

Anda dapat menentukan serangkaian rentang waktu atau ekspresi cron saat membuat jendela peristiwa, tetapi tidak keduanya.

Untuk membuat jendela peristiwa dengan rentang waktu

Gunakan perintah [create-instance-event-window](#) dan tentukan parameter `--time-range`. Anda juga tidak dapat menentukan parameter `--cron-expression`.

```
aws ec2 create-instance-event-window \
  --region us-east-1 \
  --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
  --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
  --name myEventWindowName
```

Output yang diharapkan

```
{
  "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-instance-event-window](#) dan tentukan parameter `--cron-expression`. Anda juga tidak dapat menentukan parameter `--time-range`.

```

aws ec2 create-instance-event-window \
  --region us-east-1 \
  --cron-expression "* 21-23 * * 2,3" \
  --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
  --name myEventWindowName

```

Output yang diharapkan

```

{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "State": "creating",
    "Tags": [
      {
        "Key": "K1",

```

```

    "Value": "v1"
  }
]
}
}

```

Mengaitkan target dengan jendela peristiwa

Anda hanya dapat mengaitkan satu tipe target (ID instans, ID Host Khusus, atau tanda instans) dengan suatu jendela peristiwa.

Untuk mengaitkan tanda instans dengan jendela peristiwa

Gunakan perintah [associate-instance-event-window](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk mengaitkan tanda instans, tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa tanda.

```

aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"

```

Output yang diharapkan

```

{
  "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 satu instans atau lebih dengan jendela peristiwa

Gunakan perintah [associate-instance-event-window](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk mengaitkan instans, tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID instans.

```

aws ec2 associate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"

```

Output yang diharapkan

```

{
  "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-instance-event-window](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk mengaitkan Host Khusus,

tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID Host Khusus.

```
aws ec2 associate-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

Output yang diharapkan

```
{  
  "InstanceEventWindow": {  
    "InstanceEventWindowId": "iew-0abcdef1234567890",  
    "Name": "myEventWindowName",  
    "CronExpression": "* 21-23 * * 2,3",  
    "AssociationTarget": {  
      "InstanceIds": [],  
      "Tags": [],  
      "DedicatedHostIds": [  
        "h-029fa35a02b99801d"  
      ]  
    },  
    "State": "creating"  
  }  
}
```

Memodifikasi jendela peristiwa

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.

Gunakan salah satu metode berikut untuk membuat jendela peristiwa.

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 jendela acara menggunakan AWS CLI, Anda dapat memodifikasi rentang waktu atau ekspresi cron, dan mengaitkan atau memisahkan satu atau beberapa target dengan jendela acara.

Memodifikasi waktu jendela peristiwa

Anda dapat memodifikasi rentang waktu atau ekspresi cron saat memodifikasi jendela peristiwa, tetapi tidak keduanya.

Untuk memodifikasi rentang waktu jendela peristiwa

Gunakan perintah [modify-instance-event-window](#) dan tentukan jendela peristiwanya untuk memodifikasi. Tentukan parameter `--time-range` untuk memodifikasi rentang waktu. Anda juga tidak dapat menentukan parameter `--cron-expression`.

```
aws ec2 modify-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8
```

Output yang diharapkan

```
{  
  "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](#) dan tentukan jendela peristiwanya untuk memodifikasi. Tentukan parameter `--time-range` untuk memodifikasi rentang waktu. Anda juga tidak dapat menentukan parameter `--cron-expression` dalam panggilan yang sama.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
"wednesday", "EndHour": 8},
{"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
"EndHour": 8

```

Output yang diharapkan

```

{
  "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](#) dan tentukan jendela peristiwanya untuk memodifikasi. Tentukan parameter `--cron-expression` untuk memodifikasi ekspresi cron. Anda juga tidak dapat menentukan parameter `--time-range`.

```

aws ec2 modify-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --cron-expression "* 21-23 * * 2,3"

```

Output yang diharapkan

```

{
  "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"
      }
    ]
  }
}

```

Memodifikasi target yang dikaitkan dengan jendela peristiwa

Anda dapat mengaitkan target tambahan dengan jendela peristiwa. Anda juga dapat memisahkan target yang ada dari jendela peristiwa. Namun, hanya satu tipe target (ID instans, ID Host Khusus, atau tanda instans) yang dapat dikaitkan dengan suatu jendela peristiwa.

Untuk mengaitkan target tambahan dengan jendela peristiwa

Untuk petunjuk tentang cara mengaitkan target dengan jendela peristiwa, lihat [Associate a target with an event window](#).

Untuk memisahkan tanda instans dari jendela peristiwa

Gunakan perintah [disassociate-instance-event-window](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk memisahkan tanda instans, tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa tanda.

```

aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"

```

Output yang diharapkan

```
{
  "InstanceEventWindow": {
    "InstanceEventWindowId": "iew-0abcdef1234567890",
    "Name": "myEventWindowName",
    "CronExpression": "* 21-23 * * 2,3",
    "AssociationTarget": {
      "InstanceIds": [],
      "Tags": [],
      "DedicatedHostIds": []
    },
    "State": "creating"
  }
}
```

Untuk memisahkan satu instans atau lebih dari jendela peristiwa

Gunakan perintah [disassociate-instance-event-window](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk memisahkan instans, tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID instans.

```
aws ec2 disassociate-instance-event-window \
  --region us-east-1 \
  --instance-event-window-id iew-0abcdef1234567890 \
  --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Output yang diharapkan

```
{
  "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](#) dan tentukan parameter `instance-event-window-id` untuk menentukan jendela peristiwa. Untuk memisahkan Host Khusus, tentukan parameter `--association-target`, dan untuk nilai parameternya, tentukan satu atau beberapa ID Host Khusus.

```
aws ec2 disassociate-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --association-target DedicatedHostIds=h-029fa35a02b99801d
```

Output yang diharapkan

```
{  
  "InstanceEventWindow": {  
    "InstanceEventWindowId": "iew-0abcdef1234567890",  
    "Name": "myEventWindowName",  
    "CronExpression": "* 21-23 * * 2,3",  
    "AssociationTarget": {  
      "InstanceIds": [],  
      "Tags": [],  
      "DedicatedHostIds": []  
    },  
    "State": "creating"  
  }  
}
```

Menghapus jendela peristiwa

Anda dapat menghapus satu jendela peristiwa pada satu waktu menggunakan salah satu metode berikut.

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 \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890
```

Untuk menghapus paksa jendela peristiwa

Gunakan parameter `--force-delete` jika jendela peristiwa saat ini dikaitkan dengan target.

```
aws ec2 delete-instance-event-window \  
  --region us-east-1 \  
  --instance-event-window-id iew-0abcdef1234567890 \  
  --force-delete
```

Output yang diharapkan

```
{  
  "InstanceEventWindowState": {  
    "InstanceEventWindowId": "iew-0abcdef1234567890",  
    "State": "deleting"  
  }  
}
```

Menandai jendela peristiwa

Anda dapat menandai jendela peristiwa saat membuatnya, atau setelahnya.

Untuk menandai jendela peristiwa saat membuatnya, lihat [Membuat jendela peristiwa](#).

Gunakan salah satu metode berikut untuk membuat jendela peristiwa.

Console

Untuk menandai jendela peristiwa yang ada

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, Kelola tanda jendela peristiwa instans.
5. Pilih Tambahkan tanda untuk menambahkan tanda. Ulangi hal itu untuk setiap tanda.
6. Pilih Simpan.

AWS CLI

Untuk menandai jendela peristiwa yang ada

Gunakan perintah [create-tags](#) untuk menandai sumber daya yang ada. Dalam contoh berikut, jendela peristiwa yang ada ditandai dengan Key=purpose dan Value=test.

```
aws ec2 create-tags \  
  --resources iew-0abcdef1234567890 \  
  --tags Key=purpose,Value=test
```

Pantau instans Anda menggunakan CloudWatch

Anda dapat memantau instans menggunakan Amazon CloudWatch, yang mengumpulkan dan memproses data mentah dari Amazon EC2 menjadi metrik hampir real-time yang dapat dibaca. 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 ke CloudWatch dalam periode 5 menit. Untuk mengirim data metrik untuk instans Anda CloudWatch dalam periode 1 menit, Anda dapat mengaktifkan pemantauan terperinci pada instans. Untuk informasi selengkapnya, lihat [Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans Anda](#).

Konsol Amazon EC2 menampilkan serangkaian grafik berdasarkan data mentah dari Amazon CloudWatch. Bergantung 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 CloudWatch Panduan Pengguna Amazon.

Daftar Isi

- [Alarm instans Amazon EC2](#)
- [Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans Anda](#)
- [Buat daftar CloudWatch metrik yang tersedia untuk instans Anda](#)
- [Instal dan konfigurasi CloudWatch agen menggunakan konsol Amazon EC2 untuk menambahkan metrik tambahan](#)
- [Mendapatkan statistik untuk metrik instans Anda](#)
- [Membuat grafik metrik untuk instans Anda](#)
- [Buat CloudWatch alarm untuk sebuah contoh](#)
- [Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans](#)

Alarm instans Amazon EC2

Anda dapat melihat dan membuat CloudWatch alarm Amazon untuk instans Anda dari layar Instans di konsol Amazon EC2.

Tangkapan layar berikut menunjukkan kontrol konsol, bernomor 1 dan 2, untuk melihat dan membuat alarm dari layar Instans.

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	My-1-Spot-Ins...	I-01aeed690c9fb5322	Running	t3.nano	2/2 checks passed	View alarms
<input type="checkbox"/>	My-2-Spot-Ins...	I-0ba5e5bbc9d634fa6	Stopped	t3.nano	-	View alarms

Melihat alarm dari layar Instans

Anda dapat melihat alarm setiap instans dari layar Instans.

Untuk melihat alarm instance dari layar Instances

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Dalam tabel Instances, untuk instance yang Anda pilih, pilih Lihat alarm (bernomor 1 pada tangkapan layar sebelumnya).
4. Di jendela Detail alarm untuk **i-0123456789example**, pilih nama alarm untuk melihat alarm di konsol. CloudWatch

Membuat alarm dari layar Instans

Anda dapat membuat alarm untuk setiap instance dari layar Instans.

Untuk membuat alarm untuk sebuah instance dari layar Instances

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Dalam tabel Instances, untuk contoh yang Anda pilih, pilih tanda plus (bernomor 2 pada tangkapan layar sebelumnya).
4. Di layar Kelola CloudWatch alarm, buat alarm Anda. Untuk informasi selengkapnya, lihat [Buat CloudWatch alarm untuk sebuah contoh](#).

Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans Anda

Secara default, instans Anda diaktifkan untuk pemantauan dasar. Anda secara opsional dapat mengaktifkan pemantauan terperinci.

Tabel berikut menyoroti perbedaan antara pemantauan dasar dan pemantauan terperinci untuk instans.

Tipe pemantau an	Deskripsi	Biaya
Pemantauan dasar	Hanya metrik pemeriksaan status yang tersedia dalam periode 1 menit.	Tidak dikenai biaya.

Tipe pemantauan	Deskripsi	Biaya
	Semua metrik lainnya tersedia dalam periode 5 menit.	
Pemantauan terperinci	Semua metrik, termasuk metrik pemeriksaan status, tersedia dalam periode 1 menit. Untuk mendapatkan tingkat data ini, Anda harus secara khusus mengaktifkannya untuk instans. Untuk instans yang di dalamnya Anda telah mengaktifkan pemantauan terperinci, Anda juga bisa mendapatkan data agregat di seluruh grup instans yang serupa.	Anda dikenakan biaya per metrik yang dikirim ke CloudWatch. Anda tidak dikenai biaya untuk penyimpanan data. Untuk informasi selengkapnya, lihat Tingkat berbayar dan Contoh 1 - Pemantauan Terperinci EC2 di halaman CloudWatch harga Amazon .

Topik

- [Izin IAM yang diperlukan](#)
- [Mengaktifkan pemantauan terperinci](#)
- [Menonaktifkan pemantauan terperinci](#)

Izin IAM 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](#).

Mengaktifkan pemantauan terperinci

Anda dapat mengaktifkan pemantauan terperinci pada sebuah instans saat Anda meluncurkannya atau setelah instans tersebut berjalan atau dihentikan. Mengaktifkan pemantauan terperinci pada sebuah instans tidak memengaruhi pemantauan volume EBS yang dilampirkan ke instans tersebut. Untuk informasi selengkapnya, lihat [CloudWatch Metrik Amazon untuk Amazon EBS](#).

Console

Untuk mengaktifkan pemantauan terperinci untuk instans yang ada

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 pemecahan masalah, Kelola pemantauan terperinci.
4. Pada halaman detail Pemantauan terperinci, untuk Pemantauan terperinci, pilih kotak centang Aktifkan.
5. Pilih Simpan.

Untuk mengaktifkan pemantauan terperinci saat meluncurkan suatu instans

Saat meluncurkan instans menggunakan konsol Amazon EC2, di bawah Detail lanjutan, pilih kotak centang CloudWatch Pemantauan terperinci.

AWS CLI

Untuk mengaktifkan pemantauan terperinci pada instans yang ada

Gunakan perintah [monitor-instances](#) berikut untuk mengaktifkan pemantauan terperinci pada instans yang telah ditentukan.

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

Untuk mengaktifkan pemantauan terperinci saat meluncurkan suatu instans

Gunakan perintah [run-instances](#) dengan bendera `--monitoring` untuk mengaktifkan pemantauan terperinci.

```
aws ec2 run-instances --image-id ami-09092360 --monitoring Enabled=true...
```

Menonaktifkan pemantauan terperinci

Anda dapat menonaktifkan pemantauan terperinci pada sebuah instans saat Anda meluncurkannya atau setelah instans tersebut berjalan atau dihentikan.

Console

Untuk menonaktifkan pemantauan terperinci

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 pemecahan masalah, Kelola pemantauan terperinci.
4. Pada halaman detail Pemantauan terperinci, untuk Pemantauan terperinci, kosongkan kotak centang Aktifkan.
5. Pilih Simpan.

AWS CLI

Untuk menonaktifkan pemantauan terperinci

Gunakan perintah [unmonitor-instances](#) berikut untuk menonaktifkan pemantauan terperinci pada instans yang telah ditentukan.

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

Buat daftar CloudWatch metrik yang tersedia untuk instans Anda

Amazon EC2 mengirimkan metrik ke Amazon. CloudWatch Anda dapat menggunakan, API AWS Management Console AWS CLI, atau API untuk membuat daftar metrik yang dikirimkan 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 cara mendapatkan statistik untuk metrik tersebut, lihat [Mendapatkan statistik untuk metrik instans Anda](#).

Daftar Isi

- [Metrik instans](#)
- [Metrik kredit CPU](#)
- [Metrik Host Khusus](#)

- [Metrik Amazon EBS untuk instans berbasis Nitro](#)
- [Metrik pemeriksaan status](#)
- [Metrik pencerminan lalu lintas](#)
- [Metrik grup Auto Scaling](#)
- [Dimensi metrik Amazon EC2](#)
- [Metrik penggunaan Amazon EC2](#)
- [Membuat daftar metrik menggunakan konsol](#)
- [Daftar metrik menggunakan AWS CLI](#)

Metrik instans

Namespace AWS/EC2 mencakup metrik instans berikut.

Metrik	Deskripsi	Unit	Statistik yang bermakna
CPUUtilization	<p>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.</p> <p>Pada tingkat yang sangat tinggi, CPUUtilization adalah jumlah CPUUtilization tamu dan CPUUtilization hypervisor.</p> <p>Alat dalam sistem operasi Anda dapat menunjukkan persentase yang berbeda dari CloudWatch faktor seperti simulasi perangkat lama, konfigurasi perangkat non-warisan, beban kerja interupsi berat, migrasi langsung, dan pembaruan langsung.</p>	Persen	<ul style="list-style-type: none"> • Rata-rata • Minimum • Maksimum
DiskReadOps	Operasi baca yang diselesaikan dari semua volume penyimpanan instans yang tersedia	Hitung	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
	<p>untuk instans tersebut dalam periode waktu yang ditentukan.</p> <p>Untuk menghitung rata-rata operasi I/O per detik (IOPS) selama periode tersebut, bagilah total operasi dalam periode dengan jumlah detik dalam periode tersebut.</p> <p>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan.</p>		<ul style="list-style-type: none">• Maksimum
DiskWrite Ops	<p>Operasi tulis yang diselesaikan ke semua volume penyimpanan instans yang tersedia untuk instans tersebut dalam periode waktu yang ditentukan.</p> <p>Untuk menghitung rata-rata operasi I/O per detik (IOPS) selama periode tersebut, bagilah total operasi dalam periode dengan jumlah detik dalam periode tersebut.</p> <p>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan.</p>	Hitung	<ul style="list-style-type: none">• Jumlah• Rata-rata• Minimum• Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
DiskReadBytes	<p>Bitas yang dibaca dari semua volume penyimpanan instans yang tersedia untuk instans.</p> <p>Metrik ini digunakan untuk menentukan volume data yang dibaca aplikasi dari hard disk instans. Metrik ini dapat digunakan untuk menentukan kecepatan aplikasi.</p> <p>Jumlah yang dilaporkan adalah jumlah bitas yang diterima selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi jumlah ini dengan 300 untuk menemukan Bitas/detik. 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 DiskReadBytes CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p> <p>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
DiskWrite Bytes	<p>Bitanya yang ditulis ke semua volume penyimpanan instans yang tersedia untuk instans.</p> <p>Metrik ini digunakan untuk menentukan volume data yang ditulis aplikasi ke hard disk instans. Metrik ini dapat digunakan untuk menentukan kecepatan aplikasi.</p> <p>Jumlah yang dilaporkan adalah jumlah bita yang diterima selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi jumlah ini dengan 300 untuk menemukan Bita/detik. 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 DiskWriteBytes CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p> <p>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
MetadataNoToken	<p>Berapa kali Layanan Metadata Instance (IMDS) berhasil diakses menggunakan metode yang tidak menggunakan token.</p> <p>Metrik ini digunakan untuk menentukan apakah ada proses yang mengakses metadata instance yang menggunakan Layanan Metadata Instance Versi 1 (IMDSv1), yang tidak menggunakan token. Jika semua permintaan menggunakan sesi yang didukung token, yaitu, Layanan Metadata Instans Versi 2 (IMDSv2), nilainya adalah 0. Untuk informasi selengkapnya, lihat Transisi ke penggunaan Layanan Metadata Instans Versi 2.</p>	Hitung	<ul style="list-style-type: none"> Jumlah Persentil
MetadataNoTokenRejected	<p>Berapa kali panggilan IMDSv1 dicoba setelah IMDSv1 dinonaktifkan.</p> <p>Jika metrik ini muncul, ini menunjukkan bahwa panggilan IMDSv1 telah 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.</p>	Hitung	<ul style="list-style-type: none"> Jumlah Persentil

Metrik	Deskripsi	Unit	Statistik yang bermakna
NetworkIn	<p>Jumlah bita yang diterima oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas jaringan yang masuk ke instans tunggal.</p> <p>Jumlah yang dilaporkan adalah jumlah bita yang diterima selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit) dan statistiknya adalah Sum, Anda dapat membagi jumlah ini dengan 300 untuk menemukan Bita/detik. Jika Anda memiliki pemantauan terperinci (1 menit) dan statistiknya adalah Sum, bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik DIFF_TIME untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik NetworkIn CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
NetworkOut	<p>Jumlah bita yang dikirimkan oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas jaringan yang keluar dari instans tunggal.</p> <p>Jumlah yang dilaporkan adalah jumlah bita yang dikirimkan selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit) dan statistiknya adalah Sum, Anda dapat membagi jumlah ini dengan 300 untuk menemukan Bita/detik. Jika Anda memiliki pemantauan terperinci (1 menit) dan statistiknya adalah Sum, bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik DIFF_TIME untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik NetworkOut CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
NetworkPacketsIn	<p>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.</p> <p>Metrik ini hanya tersedia untuk pemantauan dasar (periode 5 menit). Untuk menghitung jumlah paket per detik (PPS) yang diterima oleh instans Anda selama 5 menit, bagilah nilai statistik Sum dengan 300. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik DIFF_TIME untuk menemukan paket per detik. Misalnya, jika Anda telah membuat grafik NetworkPacketsIn CloudWatch sebagai $m1$, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam paket/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menganalisis metrik di Panduan CloudWatch Pengguna Amazon.</p>	Hitung	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
NetworkPacketsOut	<p>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.</p> <p>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. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik DIFF_TIME untuk menemukan paket per detik. Misalnya, jika Anda telah membuat grafik NetworkPacketsOut CloudWatch sebagaim1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam paket/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>	Hitung	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

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	Kredit (Menit vCPU)	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
	<p>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).</p> <p>Metrik kredit CPU hanya tersedia pada frekuensi 5 menit. Jika Anda menentukan periode lebih dari lima menit, gunakan statistik Sum, bukan statistik Average.</p>		<ul style="list-style-type: none">• Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
<p><code>CPUCreditBalance</code></p>	<p>Jumlah kredit CPU yang diperoleh yang diakumulasi oleh instans sejak diluncurkan atau dimulai. Untuk T2 Standar, <code>CPUCreditBalance</code> juga mencakup jumlah kredit peluncuran yang telah diakumulasi.</p> <p>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.</p> <p>Kredit dalam <code>CPUCreditBalance</code> tersedia untuk instans untuk digunakan hingga melonjak melebihi pemanfaatan CPU acuan.</p> <p>Saat sebuah instans berjalan, kredit dalam <code>CPUCreditBalance</code> tidak akan kedaluwarsa. Saat instans T3 atau T3a berhenti, nilai <code>CPUCreditBalance</code> akan bertahan selama tujuh hari. Setelah itu, semua kredit yang dikumpulkan akan hilang. Saat instans T2 berhenti, nilai <code>CPUCreditBalance</code> tidak bertahan, dan semua kredit yang masih harus dibayar hilang.</p> <p>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit.</p>	<p>Kredit (Menit vCPU)</p>	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
CPUSurplusCreditBalance	<p>Jumlah kredit surplus yang telah digunakan oleh instans unlimited saat nilai CPUCreditBalance miliknya adalah nol.</p> <p>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.</p> <p>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit.</p>	Kredit (Menit vCPU)	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum
CPUSurplusCreditsCharged	<p>Jumlah kredit surplus yang digunakan yang tidak ditutupi oleh kredit CPU yang diperoleh, sehingga menimbulkan biaya tambahan.</p> <p>Kredit surplus yang digunakan akan dikenai biaya jika salah satu dari hal berikut terjadi:</p> <ul style="list-style-type: none"> • 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. • Instans dihentikan atau diakhiri. • instans dialihkan dari unlimited ke standard. <p>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit.</p>	Kredit (Menit vCPU)	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik Host Khusus

Namespace `AWS/EC2` mencakup metrik berikut untuk Host Khusus T3.

Metrik	Deskripsi	Unit	Statistik yang bermakna
<code>DedicatedHostCPUUtilization</code>	Persentase alokasi kapasitas komputasi yang saat ini digunakan oleh instans yang berjalan di Host Khusus.	Persen	<ul style="list-style-type: none"> Jumlah Rata-rata Minimum Maksimum

Metrik Amazon EBS untuk instans berbasis Nitro

Namespace `AWS/EC2` mencakup metrik Amazon EBS tambahan untuk volume yang dipasangkan ke instans berbasis Nitro yang bukan merupakan instans bare metal.

Metrik	Deskripsi	Unit	Statistik yang bermakna
<code>EBSReadOperations</code>	<p>Operasi baca yang diselesaikan dari semua volume Amazon EBS yang dilampirkan ke instans dalam periode waktu yang ditentukan.</p> <p>Untuk menghitung rata-rata operasi I/O per detik dari pembacaan (IOPS Baca) selama periode tersebut, bagilah total operasi dalam periode dengan jumlah detik pada 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 <code>CloudWatch</code> metrik <code>DIFF_TIME</code> untuk menemukan operasi per detik. Misalnya, jika Anda telah</p>	Hitung	<ul style="list-style-type: none"> Jumlah Rata-rata Minimum Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
	<p>membuat grafik EBSReadOps CloudWatch sebagai berikut, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam operasi/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>		
EBSWriteOps	<p>Operasi tulis yang diselesaikan ke semua volume EBS yang dilampirkan pada instansi tersebut dalam periode waktu yang ditentukan.</p> <p>Untuk menghitung rata-rata operasi I/O per detik dari penulisan (IOPS Tulis) selama periode tersebut, bagilah total operasi dalam periode dengan jumlah detik pada 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 EBSWriteOps CloudWatch sebagai berikut, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam operasi/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>	Hitung	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
EBSReadBytes	<p>Bitas yang dibaca dari semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu.</p> <p>Jumlah yang dilaporkan adalah jumlah bitas baca selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Bitas/detik dari Pembacaan. 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 EBSReadBytes CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menganalisis metrik matematika di Panduan CloudWatch Pengguna Amazon.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
EBSWriteBytes	<p>Bitas yang ditulis ke semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu.</p> <p>Jumlah yang dilaporkan adalah jumlah bitas tulis selama periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi jumlah ini dengan 300 untuk menemukan Bitas/detik dari Penulisan. 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 EBSWriteBytes CloudWatch sebagai m1, rumus matematika metrik $m1 / (\text{DIFF_TIME}(m1))$ mengembalikan metrik dalam byte/detik. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menganalisis matematika metrik di Panduan CloudWatch Pengguna Amazon.</p>	Byte	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
EBSIOBalance%	<p>Memberikan informasi tentang persentase kredit I/O yang tersisa di bucket lonjakan. Metrik ini hanya tersedia untuk pemantauan dasar.</p> <p>Metrik ini hanya tersedia untuk beberapa ukuran instans <code>*.4xlarge</code> dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali.</p> <p>Statistik Sum tidak berlaku untuk metrik ini.</p>	Persen	<ul style="list-style-type: none"> • Minimum • Maksimum
EBSByteBalance%	<p>Memberikan informasi tentang persentase kredit throughput yang tersisa di bucket lonjakan. Metrik ini hanya tersedia untuk pemantauan dasar.</p> <p>Metrik ini hanya tersedia untuk beberapa ukuran instans <code>*.4xlarge</code> dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali.</p> <p>Statistik Sum tidak berlaku untuk metrik ini.</p>	Persen	<ul style="list-style-type: none"> • Minimum • 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 Spot Anda, lihat [CloudWatch metrik untuk Spot Fleet](#).

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 Anda](#).

Namespace AWS/EC2 mencakup metrik pemeriksaan status berikut.

Metrik	Deskripsi	Unit	Statistik yang bermakna
StatusCheckFailed	Melaporkan apakah instans telah melalui pemeriksaan status instans dan pemeriksaan status sistem pada menit terakhir. Metrik ini dapat berupa 0 (lulus) atau 1 (gagal). Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya.	Hitung	<ul style="list-style-type: none"> Jumlah Rata-rata
StatusCheckFailed_Instance	Melaporkan apakah instans telah melalui pemeriksaan status instan pada menit terakhir. Metrik ini dapat berupa 0 (lulus) atau 1 (gagal). Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya.	Hitung	<ul style="list-style-type: none"> Jumlah Rata-rata
StatusCheckFailed_System	Melaporkan apakah instans telah melalui pemeriksaan status sistem pada menit terakhir. Metrik ini dapat berupa 0 (lulus) atau 1 (gagal). Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya.	Hitung	<ul style="list-style-type: none"> Jumlah Rata-rata
StatusCheckFailed_AttachedEBS	Melaporkan apakah instans telah melalui pemeriksaan status EBS yang terlampir pada menit terakhir. Metrik ini dapat berupa 0 (lulus) atau 1 (gagal).	Hitung	<ul style="list-style-type: none"> Jumlah Rata-rata

Metrik	Deskripsi	Unit	Statistik yang bermakna
	Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya.		

AWS/EBSNamespace menyertakan metrik pemeriksaan status berikut.

Metrik	Deskripsi	Unit	Statistik yang bermakna
VolumeStalledIOCheck	<p>Catatan: Khusus instans Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas.</p> <p>Melaporkan apakah volume telah lulus atau gagal pemeriksaan IO yang macet di menit terakhir. Metrik ini dapat berupa 0 (lulus) atau 1 (gagal).</p>	Hitung	<ul style="list-style-type: none"> • Jumlah • Rata-rata • Minimum • Maksimum

Metrik pencerminan lalu lintas

Namespace AWS/EC2 mencakup metrik untuk lalu lintas yang dicerminkan. Untuk informasi selengkapnya, lihat [Memantau lalu lintas cermin menggunakan Amazon CloudWatch](#) di Panduan Pencerminan Lalu Lintas VPC Amazon.

Metrik grup Auto Scaling

Namespace AWS/AutoScaling mencakup metrik untuk grup Auto Scaling. Untuk informasi selengkapnya, lihat [Monitor CloudWatch metrik untuk grup dan instans Auto Scaling](#) di Panduan Pengguna Auto Scaling 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) 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 membandingkan 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 layanan. AWS 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 CloudWatch Panduan Pengguna Amazon.

Amazon EC2 menerbitkan metrik berikut di namespace AWS/Usage.

Metrik	Deskripsi
ResourceCount	<p>Jumlah sumber daya yang ditentukan yang berjalan di akun Anda. Sumber daya tersebut ditentukan oleh dimensi yang dikaitkan dengan metrik.</p> <p>Statistik yang paling berguna untuk metrik ini adalah MAXIMUM, yang merepresentasikan jumlah maksimum sumber daya yang digunakan selama periode 1 menit.</p>

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 .

Dimensi	Deskripsi
	Nilai untuk dimensi ini menentukan huruf pertama dari tipe instans yang dilaporkan oleh metrik. Misalnya, <code>Standard/OnDemand</code> mengembalikan informasi terkait semua instans yang berjalan dengan tipe yang dimulai dengan A, C, D, H, I, M, R, T, dan Z, lalu <code>G/OnDemand</code> mengembalikan informasi terkait semua instans yang berjalan dengan tipe yang dimulai dengan G.

Membuat daftar metrik menggunakan konsol

Metrik dikelompokkan berdasarkan namespace terlebih dahulu, lalu berdasarkan kombinasi dimensi dalam setiap namespace. Misalnya, Anda dapat melihat semua metrik yang disediakan oleh Amazon EC2, atau metrik yang dikelompokkan berdasarkan ID instans, tipe instans, ID citra (AMI), atau grup Auto Scaling.

Untuk melihat metrik yang tersedia berdasarkan kategori (konsol)

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi, perluas Metrik, lalu pilih Semua metrik.
3. Pilih namespace metrik EC2.

The screenshot shows the AWS CloudWatch console interface. At the top, there are navigation tabs: **Browse**, **Multi source query**, **Graphed metrics**, **Options**, and **Source**. On the right, there are buttons for **Add math** and **Add query**. Below the navigation, the main heading is **Metrics (1,153) Info**. There are several interactive elements: a toggle for **Alarm recommendations**, a **Download alarm code** button, a **Create alarm** button, **Graph with SQL** and **Graph search** buttons, a region dropdown set to **Ireland**, a search bar with the placeholder text "Search for any metric, dimension, resource id or account id", and a **Search iGraph** button.

Backup • View automatic dashboard	16	Directory Service • View automatic dashboard	62	EBS • View automatic dashboard	47
EC2 • View automatic dashboard	93	EC2/API • View automatic dashboard	152	EC2 Capacity Reservations • View automatic dashboard	8
EC2 Spot • View automatic dashboard	618	EFS • View automatic dashboard	36	Events • View automatic dashboard	1
Logs • View automatic dashboard	3	NATGateway • View automatic dashboard	15	S3 • View automatic dashboard	12
SSM Run Command • View automatic dashboard	3	Usage • View automatic dashboard	87		

4. Pilih dimensi metrik (misalnya, Metrik Per-Instans).

The screenshot shows the AWS CloudWatch console interface with filters applied. The navigation tabs are the same as in the previous screenshot. The main heading is **Metrics (93) Info**. The **Alarm recommendations** toggle is still present. The **Download alarm code** button now shows "(14)". The region dropdown is still **Ireland**. The search bar now includes the breadcrumb "All > EC2" and the same placeholder text. The **Search iGraph** button is also present.

HostId	1	Per-Instance Metrics	92
---------------	---	-----------------------------	----

5. Untuk mengurutkan metrik, gunakan judul kolom. Untuk membuat grafik sebuah metrik, pilih kotak centang di samping metrik. Untuk memfilter berdasarkan sumber daya, pilih ID sumber daya, lalu pilih Tambahkan ke pencarian. Untuk memfilter berdasarkan metrik, pilih nama metrik, lalu pilih Tambahkan ke pencarian.

The screenshot shows the AWS CloudWatch console interface. At the top, there are navigation tabs: 'Browse', 'Multi source query', 'Graphed metrics', 'Options', and 'Source'. Below these are buttons for 'Add math' and 'Add query'. The main section is titled 'Metrics (92) Info'. There are several interactive elements: a toggle for 'Alarm recommendations', a 'Download alarm code (14)' button, a 'Create alarm' button, a 'Graph with SQL' button, and a 'Graph search' button. Below these is a breadcrumb navigation: 'Ireland > All > EC2 > Per-Instance Metrics'. A search bar is present with the placeholder text 'Search for any metric, dimension, resource id or account id'. The main content is a table with columns: 'Instance name 92/92', 'Instanceid', 'Metric name', and 'Alarms'. The table lists several 'fingerprint' metrics for different instance IDs. A context menu is open over the first 'fingerprint' metric, showing options: 'Add to search', 'Exclude from search', 'Search for this only', 'Add to graph', 'Graph this metric only', 'Graph all search results', 'Graph with SQL query', 'View in Resource Health', and 'View in EC2 console'. The 'Alarms' column for all metrics shows 'No alarms'.

Daftar metrik menggunakan AWS CLI

Gunakan perintah [list-metrics](#) untuk membuat daftar CloudWatch metrik untuk instance Anda.

Untuk membuat daftar semua metrik yang tersedia pada Amazon EC2 (AWS CLI)

Contoh berikut menentukan namespace AWS/EC2 untuk melihat semua metrik pada Amazon EC2.

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

Berikut adalah contoh output:

```
{
  "Metrics": [
    {
      "Namespace": "AWS/EC2",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-1234567890abcdef0"
        }
      ]
    }
  ]
}
```

```
    ],
    "MetricName": "NetworkOut"
  },
  {
    "Namespace": "AWS/EC2",
    "Dimensions": [
      {
        "Name": "InstanceId",
        "Value": "i-1234567890abcdef0"
      }
    ],
    "MetricName": "CPUUtilization"
  },
  {
    "Namespace": "AWS/EC2",
    "Dimensions": [
      {
        "Name": "InstanceId",
        "Value": "i-1234567890abcdef0"
      }
    ],
    "MetricName": "NetworkIn"
  },
  ...
]
```

Untuk membuat daftar semua metrik yang tersedia pada sebuah instans (AWS CLI)

Contoh berikut menentukan ruang nama AWS/EC2 dan dimensi InstanceId untuk melihat hasil hanya untuk instans yang ditentukan.

```
aws cloudwatch list-metrics --namespace AWS/EC2 --dimensions  
Name=InstanceId,Value=i-1234567890abcdef0
```

Untuk membuat daftar metrik di semua instans (AWS CLI)

Contoh berikut menentukan ruang nama AWS/EC2 dan nama metrik untuk melihat hasil hanya untuk metrik yang ditentukan.

```
aws cloudwatch list-metrics --namespace AWS/EC2 --metric-name CPUUtilization
```

Instal dan konfigurasi CloudWatch agen menggunakan konsol Amazon EC2 untuk menambahkan metrik tambahan

Menginstal dan mengonfigurasi CloudWatch agen menggunakan konsol Amazon EC2 dalam versi beta untuk Amazon EC2 dan dapat berubah sewaktu-waktu.

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 melakukannya untuk Anda.

Topik ini menjelaskan bagaimana Anda dapat menggunakan konsol Amazon EC2 untuk menginstal CloudWatch agen pada instans Anda dan mengonfigurasi agen untuk memancarkan metrik yang dipilih.

Untuk langkah-langkah manual untuk proses ini, lihat [Menginstal CloudWatch agen yang menggunakan AWS Systems Manager](#) di Panduan CloudWatch Pengguna Amazon. Untuk informasi selengkapnya tentang CloudWatch agen, lihat [Mengumpulkan metrik, log, dan jejak dengan CloudWatch agen](#).

Topik

- [Prasyarat](#)
- [Cara kerjanya](#)
- [Biaya](#)
- [Instal dan konfigurasi CloudWatch agen](#)

Prasyarat

Untuk menggunakan Amazon EC2 untuk menginstal dan mengonfigurasi CloudWatch agen, Anda harus memenuhi prasyarat pengguna dan instans yang dijelaskan di bagian ini.

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:

```
{
  "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 contoh: `running`
- Sistem operasi yang didukung: Linux
- AWS Systems Manager Agen (Agen SSM): Dipasang. Dua catatan tentang Agen SSM:

- Agen SSM sudah diinstal sebelumnya di beberapa Amazon Machine Images (AMI) yang disediakan oleh AWS dan pihak ketiga 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 Pengguna](#).AWS Systems Manager
- Izin IAM untuk instance: Kebijakan AWS terkelola berikut harus ditambahkan ke peran IAM yang dilampirkan ke instance:
 - [AmazonSSM ManagedInstance Core](#) - Memungkinkan sebuah instance untuk menggunakan Systems Manager untuk menginstal dan mengkonfigurasi agen. CloudWatch
 - [CloudWatchAgentServerKebijakan](#) - Memungkinkan sebuah 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 instance yang ingin Anda tambahkan metrik, memenuhi prasyarat tertentu. Kemudian, Anda dapat menggunakan konsol Amazon EC2 untuk menginstal dan mengonfigurasi CloudWatch agen pada instans yang Anda pilih.

Pertama memenuhi [prasyarat](#)

- Anda memerlukan izin IAM yang diperlukan — Sebelum memulai, pastikan bahwa pengguna atau peran konsol Anda memiliki izin IAM yang diperlukan untuk menggunakan fitur ini.
- Instans — Untuk menggunakan fitur ini, instans EC2 Anda harus instans Linux, memiliki Agen SSM diinstal, memiliki izin IAM yang diperlukan, dan berjalan.

Kemudian Anda dapat [menggunakan fitur tersebut](#)

1. Pilih instans Anda — Di konsol Amazon EC2, Anda memilih instans untuk menginstal dan mengonfigurasi agen. CloudWatch Anda kemudian memulai proses dengan memilih Configure CloudWatch agent.

2. Validasi Agen SSM - Amazon EC2 memeriksa apakah Agen SSM diinstal dan dimulai pada setiap instance. Setiap contoh yang gagal pemeriksaan ini dikecualikan dari proses. Agen SSM digunakan untuk melakukan tindakan pada instance selama proses ini.
3. Validasi izin IAM — Amazon EC2 memeriksa bahwa setiap instans memiliki izin IAM yang diperlukan untuk proses ini. Setiap contoh yang gagal pemeriksaan ini dikecualikan dari proses. Izin IAM memungkinkan CloudWatch agen untuk mengumpulkan metrik dari instance dan berintegrasi dengan menggunakan Agen AWS Systems Manager SSM.
4. CloudWatch Agen validasi — Amazon EC2 memeriksa apakah agen diinstal dan dijalankan pada setiap instance. CloudWatch 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 ia tetap 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 instance Anda, metrik tersebut akan dihapus dari instance Anda saat proses ini selesai.
6. Perbarui konfigurasi CloudWatch agen - Amazon EC2 mengirimkan konfigurasi metrik ke agen. CloudWatch Ini adalah langkah terakhir dalam prosesnya. Jika berhasil, instans Anda dapat memancarkan data untuk metrik yang dipilih dan Amazon EC2 menghapus file konfigurasi dari Parameter Store.

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 konfigurasi 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 menimpa konfigurasi CloudWatch agen yang ada. Jika Anda tidak memilih metrik yang dipilih sebelumnya, metrik tersebut akan dihapus dari instance.

Untuk menginstal dan mengonfigurasi 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 Tindakan, Pantau dan pecahkan masalah, Konfigurasi CloudWatch agen.

Tip

Fitur ini tidak tersedia di semua Wilayah AWS. Jika CloudWatchagen Konfigurasi tidak tersedia, coba Wilayah lain.

5. Untuk setiap langkah dalam proses, baca teks konsol, lalu pilih Berikutnya.
6. Untuk menyelesaikan proses, pada langkah terakhir, pilih Lengkap.

Mendapatkan statistik untuk metrik instans Anda

Anda bisa mendapatkan statistik untuk CloudWatch metrik untuk instans Anda.

Daftar Isi

- [Gambaran umum statistik](#)
- [Mendapatkan statistik untuk instans tertentu](#)
- [Mengagregasi statistik di seluruh instans](#)
- [Mengagregasi statistik menurut grup Auto Scaling](#)
- [Mengagregasi statistik menurut AMI](#)

Gambaran umum statistik

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 dalamnya 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 $\text{Sum} / \text{SampleCount}$ 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).

Mendapatkan statistik untuk instans tertentu

Contoh berikut menunjukkan kepada Anda cara menggunakan AWS Management Console atau AWS CLI untuk menentukan pemanfaatan CPU maksimum dari instans EC2 tertentu.

Persyaratan

- Anda harus memiliki ID instans. Anda bisa mendapatkan ID instans menggunakan AWS Management Console atau perintah [describe-instances](#).
- Pemantauan dasar aktif secara default, tetapi Anda dapat mengaktifkan pemantauan terperinci. Untuk informasi selengkapnya, lihat [Mengaktifkan atau menonaktifkan pemantauan terperinci untuk instans 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.

The screenshot shows the AWS CloudWatch Metrics console interface. At the top, there are tabs for 'Browse', 'Multi source query', 'Graphed metrics', 'Options', and 'Source'. Below the tabs, there are buttons for 'Add math' and 'Add query'. The main content area is titled 'Metrics (1,153) Info'. There are several interactive elements: a toggle for 'Alarm recommendations', a 'Download alarm code' button, a 'Create alarm' button, 'Graph with SQL' and 'Graph search' buttons, a search bar with the placeholder 'Search for any metric, dimension, resource id or account id', and a 'Search iGraph' button. The main area displays a grid of metric namespaces for the 'Ireland' region. Each namespace is represented by a card with the namespace name, a count of metrics, and a link to 'View automatic dashboard'.

Namespace	Count	Additional Info
Backup	16	
Directory Service	62	
EBS	47	• View automatic dashboard
EC2	93	• View automatic dashboard
EC2/API	152	
EC2 Capacity Reservations	8	• View automatic dashboard
EC2 Spot	618	• View automatic dashboard
EFS	36	• View automatic dashboard
Events	1	• View automatic dashboard
Logs	3	• View automatic dashboard
NATGateway	15	• View automatic dashboard
S3	12	• View automatic dashboard
SSM Run Command	3	• View automatic dashboard
Usage	87	• View automatic dashboard

4. Pilih dimensi Metrik Per-Instans.

[Browse](#) | [Multi source query](#) | [Graphed metrics](#) | [Options](#) | [Source](#)

[Add math](#) ▼ | [Add query](#) ▼

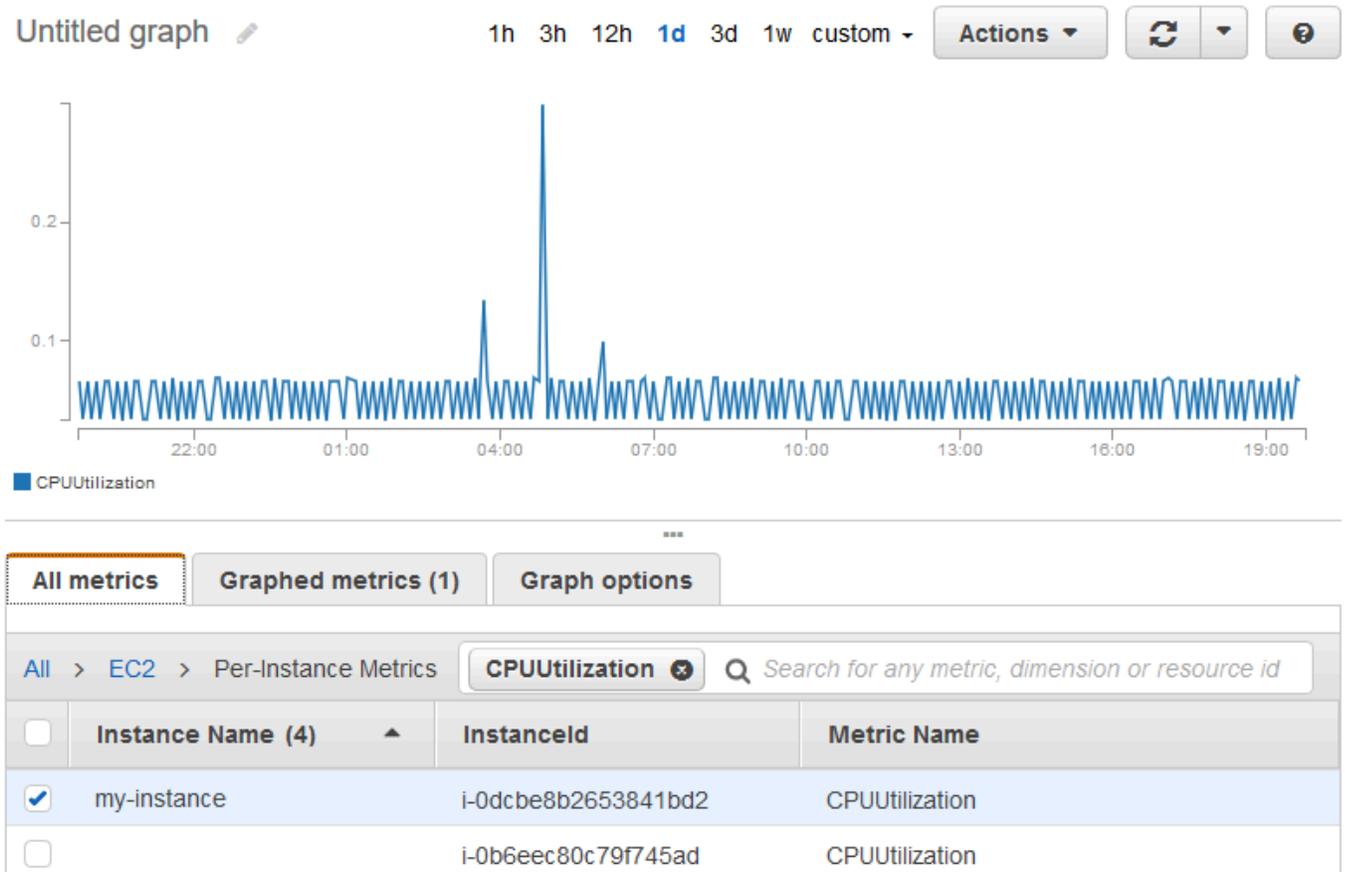
Metrics (93) [Info](#)

Alarm recommendations [Download alarm code \(14\)](#) ▼ | [Create alarm](#) | [Graph with SQL](#) | [Graph search](#)

[Ireland](#) ▼ | [All](#) > [EC2](#) |

HostId	1	Per-Instance Metrics	92
------------------------	---	--------------------------------------	----

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.

All metrics		Graphed metrics (1)		Graph options		
	Label	Namespace	Dimensions	Metric Name	Statistic <input type="checkbox"/>	Period <input type="checkbox"/>
●	CPUUtilization	EC2	Dimensions (1)	CPUUtilization	Average	<ul style="list-style-type: none"> 1 Minute 5 Minutes 15 Minutes 1 Hour 6 Hours 1 Day

Untuk mendapatkan pemanfaatan CPU pada instans 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 menggabungkan 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 di AWS/EC2 namespace.

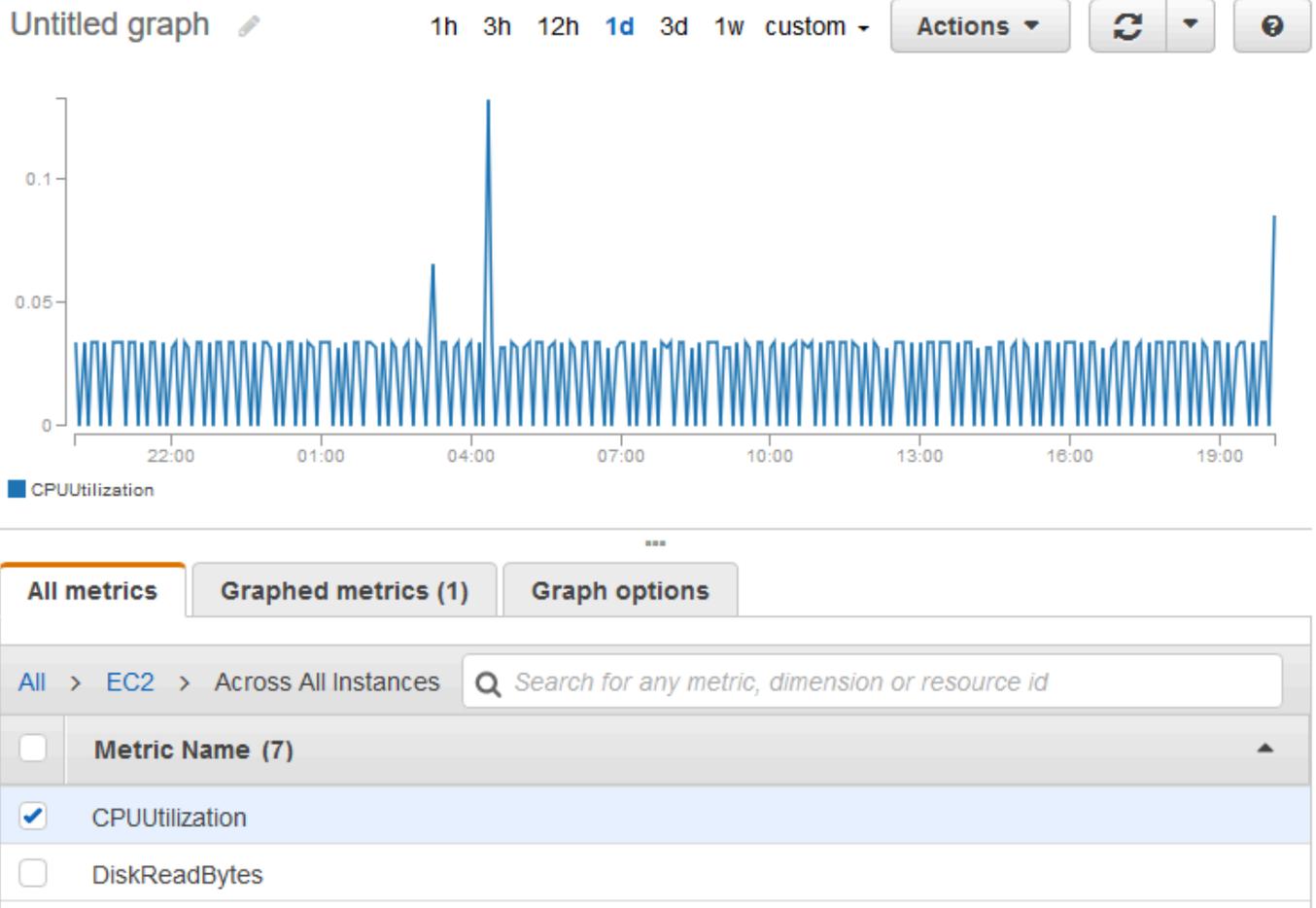
Important

Teknik untuk mengambil semua dimensi di seluruh AWS namespace ini 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.

- 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.



- Untuk mengubah statistik atau periode metrik, pilih tab Metrik grafik. Pilih judul kolom atau nilai individu, lalu pilih nilai yang berbeda.

Untuk mendapatkan rata-rata pemanfaatan CPU 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 menggabungkan 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 instance dalam grup Auto Scaling (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 untuk instance 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 instance dalam grup Auto Scaling ()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 mengagregasi statistik untuk instans Anda 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 menggabungkan 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 rata-rata pemanfaatan CPU pada ID citra (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"
}
```

Membuat grafik metrik 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 [Buat daftar CloudWatch metrik yang tersedia untuk instans Anda](#).

Metrik grafik menggunakan konsol CloudWatch

Anda juga dapat menggunakan CloudWatch konsol untuk membuat grafik data metrik yang dihasilkan oleh Amazon EC2 dan layanan lainnya AWS . Untuk informasi selengkapnya, lihat [Metrik grafik](#) di CloudWatch Panduan Pengguna Amazon.

Buat CloudWatch alarm untuk sebuah contoh

Anda dapat membuat CloudWatch alarm yang memantau CloudWatch metrik untuk salah satu instans Anda. CloudWatch akan secara otomatis mengirimkan 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 konsol. CloudWatch

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 Actions, Monitor dan troubleshoot, Manage CloudWatch alarm.
4. Pada halaman Kelola detail CloudWatch 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 \geq 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 konsol CloudWatch. Jika Anda ingin menghapus alarm Anda, Anda dapat melakukannya dari CloudWatch konsol. Untuk informasi selengkapnya, lihat [Mengedit atau menghapus CloudWatch alarm](#) di Panduan CloudWatch Pengguna Amazon.

Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans

Dengan menggunakan tindakan CloudWatch alarm Amazon, Anda dapat membuat alarm yang secara otomatis menghentikan, mengakhiri, 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 Pengguna Amazon CloudWatch](#).

Peran `AWSServiceRoleForCloudWatchEvents` terkait layanan memungkinkan AWS untuk melakukan tindakan alarm atas nama Anda. Pertama kali Anda membuat alarm di AWS Management Console, API AWS CLI, atau IAM, CloudWatch membuat peran 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 mengganggu (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 instans 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 berhenti, menghentikan, reboot, atau memulihkan ke alarm apa pun yang disetel pada metrik per instans Amazon EC2, termasuk metrik pemantauan dasar dan terperinci yang disediakan oleh CloudWatch Amazon (di AWS/EC2 namespace), serta metrik kustom apa pun yang menyertakan InstanceId dimensi, selama nilainya mengacu pada instans Amazon EC2 yang berjalan valid.

Important

Alarm pemeriksaan status dapat memasuki INSUFFICIENT_DATA status sementara jika ada titik data metrik yang hilang. Meskipun jarang, ini bisa terjadi ketika ada gangguan dalam sistem pelaporan metrik, bahkan ketika sebuah instance sehat. Sebaiknya Anda memperlakukan INSUFFICIENT_DATA status sebagai data yang hilang, bukan 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, mem-boot ulang, atau memulihkan instance](#) di CloudWatch Panduan 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 [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.

Daftar Isi

- [Tambahkan tindakan berhenti ke CloudWatch alarm Amazon](#)
- [Tambahkan tindakan penghentian ke alarm Amazon CloudWatch](#)
- [Tambahkan tindakan reboot ke CloudWatch alarm Amazon](#)

- [Tambahkan tindakan pemulihan ke CloudWatch alarm Amazon](#)
- [Gunakan CloudWatch konsol Amazon untuk melihat alarm dan riwayat tindakan](#)
- [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 perangkat root dapat dihentikan atau diakhiri, sedangkan instans yang menggunakan penyimpanan instans sebagai perangkat root hanya dapat diakhiri. 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 Actions, Monitor dan troubleshoot, Manage 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 pesan application-to-person \(A2P\) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).
 - 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 \leq 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 alarm Amazon CloudWatch

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 [Aktifkan perlindungan pengakhiran](#).

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 Actions, Monitor dan troubleshoot, Manage 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 pesan application-to-person \(A2P\) di Panduan Pengembangan Layanan Pemberitahuan Sederhana Amazon](#).
- 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 [Menyalakan ulang instans 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 [Mengevaluasi 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 Actions, Monitor dan troubleshoot, Manage 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 pesan](#)

[application-to-person \(A2P\) di Panduan Pengembang Layanan](#) Pemberitahuan Sederhana Amazon.

- 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 dari 5 Menit.
- 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 instance menjadi terganggu karena kegagalan perangkat keras yang mendasarinya atau masalah yang memerlukan AWS keterlibatan untuk memperbaiki, Anda dapat memulihkan instance 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 di 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 konfigurasi 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 [Ketahanan contoh](#).

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 [Mengevaluasi 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 Actions, Monitor dan troubleshoot, Manage 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 pesan application-to-person \(A2P\) di Panduan Pengembang Layanan](#) Pemberitahuan Sederhana Amazon.

 Note

Pengguna harus berlangganan topik SNS yang ditentukan untuk menerima notifikasi email saat alarm dipicu. Pengguna root akun AWS Selalu menerima pemberitahuan email ketika tindakan pemulihan instans otomatis terjadi, bahkan jika topik SNS tidak ditentukan atau pengguna root tidak berlangganan ke 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 dari 5 Menit.
- 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.

Gunakan CloudWatch konsol Amazon untuk melihat alarm dan riwayat tindakan

Anda dapat melihat alarm dan riwayat tindakan di CloudWatch konsol Amazon. Amazon CloudWatch menyimpan alarm dan riwayat aksi selama dua minggu terakhir.

Untuk melihat riwayat alarm dan tindakan yang dipicu (CloudWatch konsol)

1. Buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi, pilih Alarm.
3. Pilih alarm.
4. Tab Detail menunjukkan transisi status terbaru beserta nilai waktu dan metriknya.
5. Pilih tab Riwayat untuk melihat entri riwayat terbaru.

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.

New console

The screenshot displays the configuration interface for a new Amazon CloudWatch alarm. It is divided into three main sections: Alarm notification, Alarm action, and Alarm thresholds. Each section has a toggle switch and an 'Info' link. The Alarm notification section includes a search box for an existing SNS topic. The Alarm action section features a dropdown menu for selecting an action. The Alarm thresholds section contains several input fields and dropdown menus for configuring the alarm's logic, including grouping samples, data sampling type, alarm conditions, consecutive periods, and the alarm name.

Alarm notification [Info](#)

Configure the alarm to send notifications to an Amazon SNS topic when it is triggered.

1

Alarm action [Info](#)

Specify the action to take when the alarm is triggered.

Alarm thresholds

Specify the metric thresholds for the alarm.

Group samples by age

Type of data to sample

Alarm When

Consecutive Period Period nutes

Alarm name

Old console

Create Alarm ✕

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.
To edit an alarm, first choose whom to notify and then define when the notification should be sent.

1 **Send a notification to:** [create topic](#)

Take the action:

- Recover this instance (i)
- Stop this instance (i)
- Terminate this instance (i)
- Reboot this instance (i)

Whenever: **2** of **3**

Is: **4** **5** Percent

For at least: **6** consecutive period(s) of **7**

Name of alarm:

Cancel
Create Alarm

CPU Utilization Percent

Skenario 1: Menghentikan instans pengembangan dan pengujian yang mengganggu

Buat alarm yang menghentikan instans yang digunakan untuk pengembangan atau pengujian perangkat lunak saat sedang mengganggu selama setidaknya satu jam.

Pengaturan	Nilai
1	Berhenti
2	Maksimum
3	Pemanfaatan CPU
4	<=
5	10%
6	1
7	1 Jam

Skenario 2: Menghentikan instans yang mengganggu

Buat alarm yang menghentikan sebuah instans dan mengirimkan email saat instans tersebut sudah mengganggu 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

Pengaturan	Nilai
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	-

Pengaturan	Nilai
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
7	5 Menit

Otomatiskan Amazon EC2 menggunakan 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
- Menginvokasi 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 instans Amazon EC2 menggunakan Amazon EC2 Run Command setiap kali peristiwa tertentu di layanan lain terjadi. AWS

Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#).

Tipe peristiwa Amazon EC2

Amazon EC2 mendukung tipe peristiwa berikut:

- [Perubahan Status AMI EC2](#)
- [Notifikasi State-change Peluncuran Cepat EC2](#)
- [Kesalahan Armada EC2](#)
- [Informasi Armada EC2](#)
- [Perubahan Instans Armada EC2](#)
- [Perubahan Permintaan Instans Spot Armada EC2](#)
- [Perubahan Status Armada EC2](#)
- [Rekomendasi Penyeimbangan Ulang Instans EC2](#)
- [Notifikasi Perubahan Status Instans EC2](#)
- [Kesalahan Armada Spot EC2](#)
- [Informasi Armada Spot EC2](#)
- [Perubahan Instans Armada Spot EC2](#)
- [Perubahan Permintaan Instans Spot Armada Spot EC2](#)

- [Perubahan Status Armada Spot EC2](#)
- [Peringatan Interupsi Instans Spot EC2](#)
- [Pemenuhan Permintaan Instans Spot EC2](#)
- [Notifikasi Kurangnya Pemanfaatan ODCR EC2](#)

Untuk informasi tentang jenis acara yang didukung oleh Amazon EBS, lihat [EventBridge 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 untuk Amazon EC2 sebagai peristiwa, termasuk panggilan dari konsol dan dari panggilan kode ke operasi API. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke Amazon EC2 API, alamat IP dari mana permintaan dibuat, kapan dibuat, dan sebagainya.

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

Informasi API Amazon EC2 di CloudTrail

CloudTrail diaktifkan pada Akun AWS saat Anda membuat akun. Ketika aktivitas terjadi di Amazon EC2 dan Amazon EBS, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa lain dalam riwayat Layanan AWS Acara. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di Akun AWS Anda. Untuk informasi selengkapnya, lihat [Melihat peristiwa dengan Riwayat CloudTrail acara](#).

Buat jejak untuk catatan peristiwa yang sedang berlangsung di Akun AWS Anda, termasuk peristiwa untuk Amazon EC2 dan Amazon EBS. Jejak memungkinkan CloudTrail untuk mengirimkan 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 AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat:

- [Membuat jejak untuk Anda Akun AWS](#)
- [Layanan AWS integrasi dengan log CloudTrail](#)

- [Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail](#)
- [Menerima file CloudTrail log dari beberapa Wilayah](#) dan [Menerima file CloudTrail log dari beberapa akun](#)

Semua tindakan Amazon EC2, dan tindakan manajemen Amazon EBS, dicatat oleh CloudTrail dan didokumentasikan dalam Referensi API [Amazon](#) EC2. Misalnya, panggilan ke [RunInstances](#), [DescribeInstances](#), atau [CreateImage](#) tindakan menghasilkan entri dalam file CloudTrail log.

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas tersebut membantu Anda menentukan hal berikut:

- Apakah permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna IAM.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

Untuk informasi lebih lanjut, lihat [CloudTrailuserIdentityelemen](#).

Memahami entri file log API Amazon EC2

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. Peristiwa mewakili permintaan tunggal dari sumber mana pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga file tersebut 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"
}
]
```

Digunakan AWS CloudTrail untuk mengaudit koneksi yang dibuat menggunakan EC2 Instance Connect

Gunakan AWS CloudTrail untuk mengaudit pengguna yang terhubung ke instans Anda melalui EC2 Instance Connect.

Untuk mengaudit aktivitas SSH melalui EC2 Instance Connect menggunakan konsol AWS CloudTrail

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. Bidang `requestParameters` berisi ID instans tujuan, nama pengguna OS, dan kunci publik yang digunakan untuk membuat koneksi SSH.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGONGNOM00CB6XYTQEXAMPLE",
    "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 ABCDEFGHIJKLMN001234567890EXAMPLE"
  }
},
"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 acara 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.

Pantau aplikasi.NET dan SQL Server Anda dengan Application CloudWatch Insights

CloudWatch [Application Insights](#) membantu Anda memantau aplikasi.NET dan SQL Server yang menggunakan instans Amazon EC2 bersama dengan sumber daya aplikasi lainnya. Wawasan Aplikasi CloudWatch mengidentifikasi dan menyiapkan metrik, log, dan alarm kunci di seluruh sumber daya aplikasi dan tumpukan teknologi (misalnya, basis data Microsoft SQL Server, server web (IIS) dan aplikasi, OS, penyeimbang beban, dan antrean). Wawasan Aplikasi CloudWatch terus memantau metrik dan log untuk mendeteksi serta menghubungkan anomali dan kesalahan. Ketika kesalahan dan anomali terdeteksi, Application Insights menghasilkan [CloudWatch 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.

Untuk melihat daftar lengkap log dan metrik yang didukung, lihat [Log dan Metrik yang Didukung oleh Wawasan CloudWatch Aplikasi Amazon](#).

Informasi yang diberikan terkait masalah yang terdeteksi:

- Ringkasan singkat masalah
- Waktu dan tanggal mulai masalah
- Tingkat keparahan masalah: Tinggi/Medium/Rendah
- Status masalah yang terdeteksi: Sedang Berlangsung/Terselesaikan
- Wawasan: Secara otomatis menghasilkan wawasan terkait masalah yang terdeteksi dan kemungkinan akar masalah
- Umpan balik tentang wawasan: Umpan balik yang Anda berikan tentang kegunaan wawasan yang dihasilkan oleh Wawasan CloudWatch Aplikasi 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 menentukannya 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 [Wawasan CloudWatch Aplikasi](#) di Panduan CloudWatch Pengguna Amazon.

Lacak penggunaan Tingkat Gratis Anda untuk Amazon EC2

Anda dapat menggunakan Amazon EC2 tanpa dikenakan biaya jika Anda telah menjadi AWS pelanggan kurang dari 12 bulan dan Anda tetap dalam batas penggunaan. AWS Tingkat Gratis Penting untuk melacak penggunaan Tingkat Gratis Anda guna menghindari tagihan yang tidak terduga. Jika Anda melebihi batas Tingkat Gratis, Anda akan dikenakan pay-as-go biaya standar.

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).

EC2 Free Tier [Info](#)

Offers for all AWS Regions.

3 EC2 free tier offers in use

End of month forecast
⚠️ 2 offers forecasted to exceed free tier limit.

Exceeds free tier
⚠️ 1 offers exceeded and is now pay-as-you-go pricing.

[View Global EC2 resources](#)

Offer usage (monthly)

Windows EC2 Instances	<div style="width: 12%;"><div style="width: 12%;"></div></div>	12%
662 hours remaining		
Linux EC2 Instances	<div style="width: 100%;"><div style="width: 100%;"></div></div>	100%
⚠️ Offer limit reached		
Storage space on EBS	<div style="width: 85%;"><div style="width: 85%;"></div></div>	85%
4.59 GB remaining		

[View all AWS Free Tier offers](#) [↗](#)

- 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 instruksi penghapusan instans Anda, buka langkah berikutnya dalam tutorial ini.
 - 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 [Amazon EC2 Global View](#).
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 [Menggunakan AWS Tingkat Gratis](#) di Panduan Pengguna Penagihan AWS .

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. Instans menerima alamat IP privat primer dari alamat IPv4 subnet, dan ditetapkan ke antarmuka jaringan primer.

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](#)
- [Tipe nama host instans Amazon EC2](#)
- [Bring your own IP addresses \(BYOIP\) di Amazon EC2](#)
- [Alamat IP elastis](#)
- [Antarmuka jaringan elastis](#)
- [Bandwidth jaringan instans Amazon EC2](#)
- [Jaringan yang disempurnakan di Amazon EC2](#)
- [Elastic Fabric Adapter](#)

- [Topologi instans Amazon EC2](#)
- [Grup penempatan](#)
- [Maximum transmission unit \(MTU\) jaringan untuk instans EC2 Anda](#)
- [Virtual private cloud untuk instans EC2 Anda](#)

Wilayah dan Zona

Amazon EC2 dihosting di beberapa lokasi di seluruh dunia. Lokasi ini terdiri dari Wilayah AWS, Availability Zones, Local Zones AWS Outposts, dan Wavelength Zones.

- Setiap Wilayah adalah area geografis yang terpisah.
- Zona Ketersediaan adalah beberapa lokasi terisolasi di setiap Wilayah.
- Local Zones memberi Anda kemampuan untuk menempatkan sumber daya, seperti komputasi dan penyimpanan, di beberapa lokasi yang lebih dekat dengan pengguna akhir Anda.
- AWS Outposts membawa AWS layanan asli, infrastruktur, dan model operasi ke hampir semua pusat data, ruang co-lokasi, atau fasilitas lokal.
- Wavelength Zone memungkinkan developer membangun aplikasi yang menghadirkan latensi sangat rendah ke perangkat 5G dan pengguna akhir. Wavelength menyebarkan layanan komputasi dan penyimpanan AWS standar ke tepi jaringan 5G operator telekomunikasi.

AWS beroperasi state-of-the-art, pusat data yang 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 membantu Anda menentukan deployment mana yang terbaik untuk Anda, lihat [FAQ AWS Wavelength](#).

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 Anda melihat sumber daya Anda, Anda hanya melihat sumber daya yang terkait dengan Wilayah yang Anda tentukan. Ini karena Wilayah terisolasi satu sama lain, dan kami tidak secara otomatis mereplikasi sumber daya di seluruh Wilayah.

Saat Anda meluncurkan sebuah instans, Anda harus memilih AMI yang berada di Wilayah yang sama. Jika AMI berada di Wilayah lain, Anda dapat menyalin AMI ke Wilayah yang Anda gunakan. Untuk informasi selengkapnya, lihat [Menyalin AMI](#).

Perhatikan bahwa ada biaya untuk transfer data antar Wilayah. Untuk informasi selengkapnya, lihat [Harga Amazon EC2 - Transfer Data](#).

Daftar Isi

- [Wilayah yang Tersedia](#)
- [Wilayah dan titik akhir](#)
- [Menjelaskan Wilayah Anda](#)
- [Dapatkan nama tampilan Wilayah](#)
- [Menentukan Wilayah untuk sumber daya](#)

Wilayah yang Tersedia

Akun Anda menentukan Wilayah yang tersedia untuk Anda.

- Akun AWS menyediakan beberapa Wilayah sehingga Anda dapat meluncurkan instans Amazon EC2 di lokasi yang memenuhi persyaratan Anda. Misalnya, Anda mungkin ingin meluncurkan instans di Eropa agar lebih dekat dengan pelanggan Eropa Anda atau untuk memenuhi persyaratan hukum.
- Akun AWS GovCloud (AS-Barat) menyediakan akses ke Wilayah AWS GovCloud (AS-Barat) dan Wilayah AWS GovCloud (AS-Timur). Untuk informasi selengkapnya, lihat [AWS GovCloud \(US\)](#).
- Akun Amazon AWS (China) hanya menyediakan akses ke Wilayah Beijing dan Ningxia. Untuk informasi selengkapnya, lihat [Amazon Web Services di Tiongkok](#).

Tabel berikut mencantumkan Wilayah yang disediakan oleh Akun AWS. Anda tidak dapat menjelaskan atau mengakses Wilayah tambahan dari Akun AWS, seperti AWS GovCloud (US) Regions atau Wilayah China. Untuk menggunakan Wilayah yang diperkenalkan setelah 20 Maret 2019, Anda harus mengaktifkan Wilayah tersebut. Untuk informasi selengkapnya, lihat [Menentukan AWS Wilayah mana yang dapat digunakan akun Anda](#) dalam Panduan AWS Account Management Referensi.

Code	Nama	Status keikutsertaan
us-east-2	AS Timur (Ohio)	Tidak diperlukan
us-east-1	AS Timur (Virginia)	Tidak diperlukan
us-west-1	AS Barat (California Utara)	Tidak diperlukan
us-west-2	AS Barat (Oregon)	Tidak diperlukan
af-south-1	Afrika (Cape Town)	Yg dibutuhkan
ap-east-1	Asia Pasifik (Hong Kong)	Diperlukan
ap-south-2	Asia Pasifik (Hyderabad)	Diperlukan
ap-southeast-3	Asia Pasifik (Jakarta)	Diperlukan
ap-southeast-4	Asia Pasifik (Melbourne)	Diperlukan
ap-south-1	Asia Pasifik (Mumbai)	Tidak diperlukan
ap-northeast-3	Asia Pasifik (Osaka)	Tidak diperlukan
ap-northeast-2	Asia Pasifik (Seoul)	Tidak diperlukan
ap-southeast-1	Asia Pasifik (Singapura)	Tidak diperlukan
ap-southeast-2	Asia Pasifik (Sydney)	Tidak diperlukan
ap-northeast-1	Asia Pasifik (Tokyo)	Tidak diperlukan
ca-central-1	Kanada (Pusat)	Tidak diperlukan

Code	Nama	Status keikutsertaan
ca-west-1	Kanada Barat (Calgary)	Diperlukan
eu-central-1	Eropa (Frankfurt)	Tidak diperlukan
eu-west-1	Eropa (Irlandia)	Tidak diperlukan
eu-west-2	Eropa (London)	Tidak diperlukan
eu-south-1	Eropa (Milan)	Diperlukan
eu-west-3	Eropa (Paris)	Tidak diperlukan
eu-south-2	Eropa (Spanyol)	Diperlukan
eu-north-1	Eropa (Stockholm)	Tidak diperlukan
eu-central-2	Eropa (Zürich)	Diperlukan
il-central-1	Israel (Tel Aviv)	Diperlukan
me-south-1	Timur Tengah (Bahrain)	Diperlukan
me-central-1	Timur Tengah (UEA)	Diperlukan
sa-east-1	Amerika Selatan (Sao Paulo)	Tidak diperlukan

Untuk informasi lebih lanjut, lihat [AWS Infrastruktur Global](#).

Jumlah dan pemetaan Zona Ketersediaan per Wilayah dapat bervariasi antara Akun AWS Untuk mencantumkan Zona Ketersediaan yang tersedia untuk akun Anda, Anda dapat menggunakan konsol Amazon EC2 atau antarmuka baris perintah. Untuk informasi selengkapnya, lihat [Menjelaskan Wilayah Anda](#).

Wilayah dan titik akhir

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 dan kuota Amazon EC2](#) di Referensi Umum Amazon Web Services.

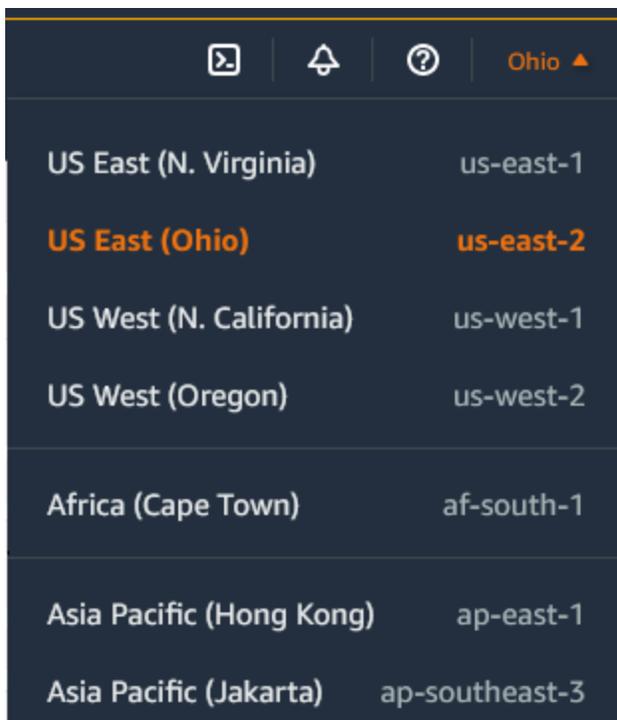
Untuk informasi selengkapnya tentang titik akhir dan protokol di AWS GovCloud (AS-Barat), lihat [Titik Akhir Layanan](#) di Panduan Pengguna.AWS GovCloud (US)

Menjelaskan Wilayah Anda

Anda dapat menggunakan konsol Amazon EC2 atau antarmuka baris perintah untuk menentukan Wilayah mana yang tersedia untuk akun Anda. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

Untuk menemukan Wilayah Anda menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari bilah navigasi, pilih pemilih Wilayah.



3. Sumber daya EC2 Anda untuk Wilayah yang dipilih ditampilkan di Dasbor EC2 di bagian Sumber Daya.

Untuk menemukan Wilayah Anda menggunakan AWS CLI

Gunakan perintah [describe-regions](#) sebagai berikut untuk mendeskripsikan Wilayah yang diaktifkan untuk akun Anda.

```
aws ec2 describe-regions
```

Untuk mendeskripsikan semua Wilayah, termasuk Wilayah yang dinonaktifkan untuk akun Anda, tambahkan opsi `--all-regions` sebagai berikut.

```
aws ec2 describe-regions --all-regions
```

Dapatkan nama tampilan Wilayah

Anda dapat menggunakan AWS Systems Manager Parameter Store untuk melihat nama tampilan Wilayah. Setiap Wilayah memiliki parameter publik di jalur berikut.

```
/aws/service/global-infrastructure/regions/region-code
```

Parameter publik untuk suatu Wilayah meliputi:

- `/aws/service/global-infrastructure/regions/region-code/domain`
- `/aws/service/global-infrastructure/regions/region-code/geolocationCountry`
- `/aws/service/global-infrastructure/regions/region-code/geolocationRegion`
- `/aws/service/global-infrastructure/regions/region-code/longName`
- `/aws/service/global-infrastructure/regions/region-code/partition`

Parameter `longName` berisi nama tampilan Wilayah. Perintah [get-parameters-by-path](#) berikut mengembalikan nama tampilan Wilayah `af-south-1`. Perintah ini menggunakan opsi `--query` untuk mencakup output ke nama Wilayah. Anda harus menempatkan string kueri dalam tanda kutip tunggal di Linux. Untuk menjalankan perintah ini menggunakan Windows Command Prompt, hilangkan tanda kutip tunggal atau ubah menjadi tanda kutip ganda.

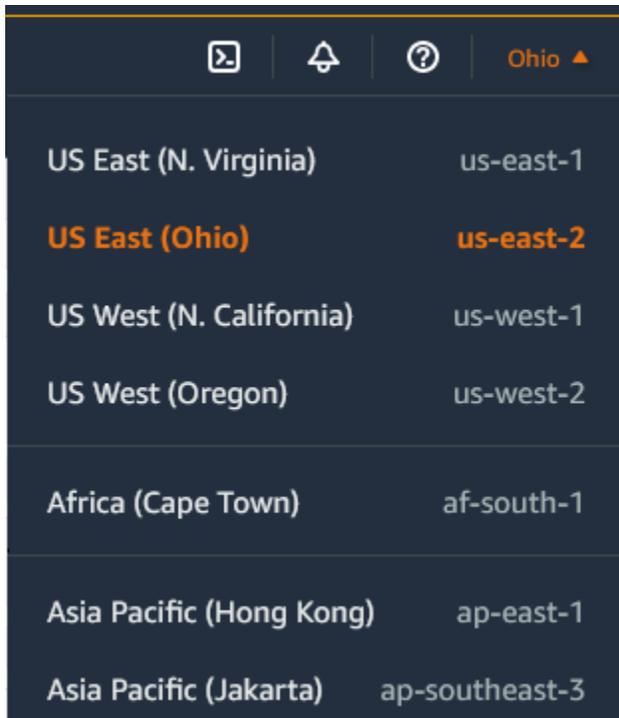
AWS CLI on Linux

```
aws ssm get-parameters-by-path \  
  --path /aws/service/global-infrastructure/regions/af-south-1 \  
  --query 'Parameters[?Name.contains(@, `longName`)].Value' \  
  --output text
```

AWS CLI on Windows

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


2. Dari bilah navigasi, pilih pemilih Wilayah, kemudian pilih Wilayah.



Untuk menentukan Area default menggunakan baris perintah

Anda dapat menetapkan nilai variabel lingkungan ke titik akhir Wilayah yang diinginkan (misalnya, `https://ec2.us-east-2.amazonaws.com`):

- `AWS_DEFAULT_REGION` (AWS CLI)
- `Set-AWSDefaultRegion` (AWS Tools for Windows PowerShell)

Sebagai alternatif, Anda dapat menggunakan opsi baris perintah `--region` (AWS CLI) atau `-Region` (AWS Tools for Windows PowerShell) dengan setiap perintah individual. Sebagai contoh, `--region us-east-2`.

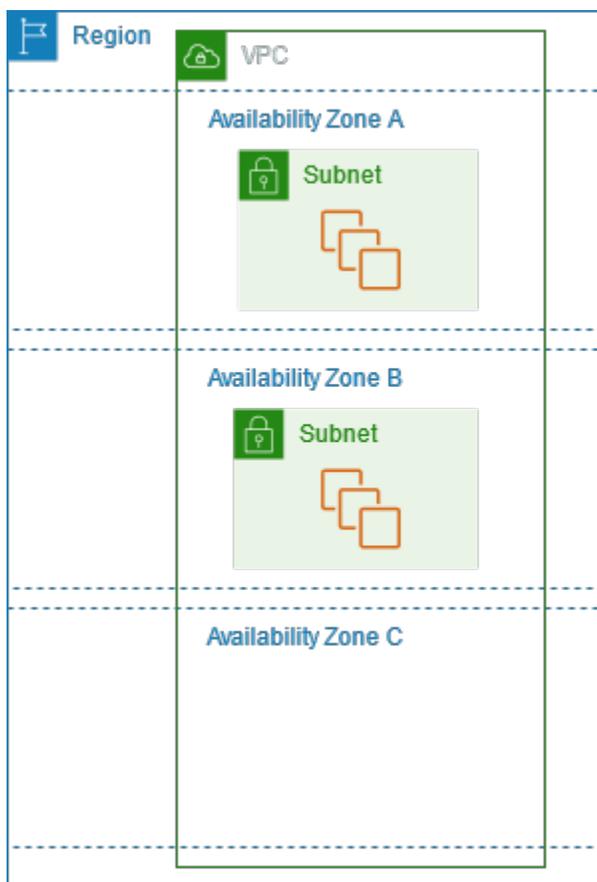
Untuk informasi selengkapnya tentang titik akhir Amazon EC2, lihat titik [akhir dan kuota Amazon EC2](#) di. Referensi Umum AWS

Zona Ketersediaan

Setiap Wilayah memiliki beberapa lokasi terisolasi yang dikenal sebagai Zona Ketersediaan. Kode untuk Zona Ketersediaan adalah kode Wilayah diikuti oleh pengidentifikasi huruf. Misalnya, `us-east-1a`.

Saat meluncurkan instans, Anda memilih Wilayah dan cloud privat virtual (VPC), kemudian Anda dapat memilih subnet dari salah satu Zona Ketersediaan atau membiarkan kami memilihkannya untuk Anda. Jika Anda mendistribusikan instans Anda ke beberapa Zona Ketersediaan dan satu instans gagal, Anda dapat mendesain aplikasi Anda sehingga instans di Zona Ketersediaan lain dapat menangani permintaan. Anda juga dapat menggunakan alamat IP Elastis untuk menutupi kegagalan instans di satu Zona Ketersediaan dengan memetakan ulang alamat secara cepat ke instans di Zona Ketersediaan lain.

Diagram berikut menggambarkan beberapa Availability Zone di suatu AWS Region. Zona Ketersediaan A dan Zona Ketersediaan B masing-masing memiliki satu subnet, dan setiap subnet memiliki instans. Zona Ketersediaan C tidak memiliki subnet, oleh karena itu Anda tidak dapat meluncurkan instans ke Zona Ketersediaan ini.



Seiring dengan berkembangnya Zona Ketersediaan dari waktu ke waktu, kemampuan kami untuk mengembangkannya dapat menjadi terbatas. Jika ini terjadi, kami mungkin membatasi Anda untuk meluncurkan sebuah instans di Zona Ketersediaan yang dibatasi kecuali Anda sudah memiliki instans di Zona Ketersediaan tersebut. Akhirnya, kami mungkin juga menghapus Zona Ketersediaan yang dibatasi dari daftar Zona Ketersediaan untuk akun baru. Oleh karena itu, akun Anda mungkin memiliki jumlah Zona Ketersediaan yang berbeda di suatu Wilayah dengan akun lain.

Daftar Isi

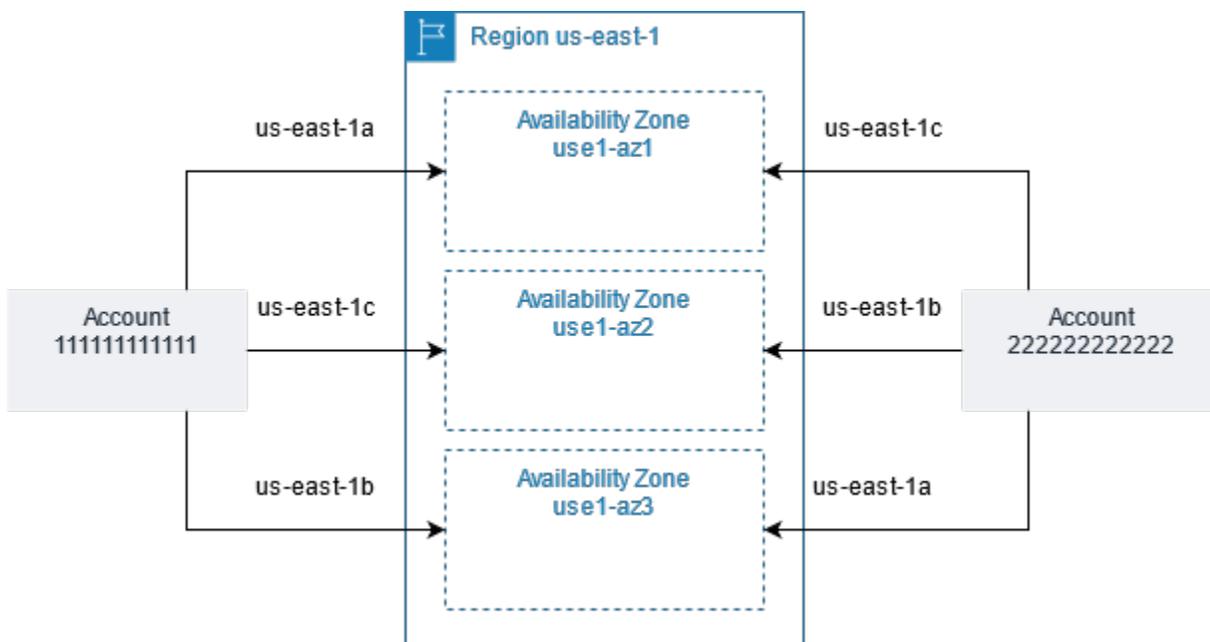
- [ID AZ](#)
- [Menjelaskan Zona Ketersediaan Anda](#)
- [Meluncurkan instans di Zona Ketersediaan](#)
- [Memigrasi sebuah instans ke Zona Ketersediaan lain](#)

ID AZ

Untuk memastikan bahwa sumber daya didistribusikan di seluruh Availability Zone untuk suatu Wilayah, kami secara independen memetakan Availability Zone ke kode untuk masing-masing Akun AWS di Wilayah tertentu kami. Misalnya, `us-east-1a` untuk Anda Akun AWS mungkin bukan lokasi fisik yang sama dengan `us-east-1a` yang lain Akun AWS.

Untuk mengoordinasikan Availability Zone di seluruh akun di semua Wilayah bahkan yang memetakan Availability Zone, gunakan ID AZ, yang merupakan pengidentifikasi unik dan konsisten untuk Availability Zone. Misalnya, `use1-az1` adalah ID AZ untuk `us-east-1` Wilayah, dan memiliki lokasi fisik yang sama di setiap wilayah Akun AWS. Anda dapat melihat ID AZ untuk akun Anda untuk menentukan lokasi fisik sumber daya Anda relatif terhadap sumber daya di akun lain. Misalnya, jika Anda membagikan subnet di Zona Ketersediaan dengan ID AZ `use1-az2` dengan akun lain, subnet ini tersedia untuk akun tersebut di Zona Ketersediaan yang juga memiliki ID AZ yang juga `use1-az2`.

Diagram berikut menggambarkan dua akun dengan pemetaan kode Zona Ketersediaan yang berbeda ke ID AZ.



Menjelaskan Zona Ketersediaan Anda

Anda dapat menggunakan konsol Amazon EC2 atau antarmuka baris perintah untuk menentukan Zona Ketersediaan mana yang tersedia untuk akun Anda. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

Untuk menemukan Zona Ketersediaan Anda 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. Di panel navigasi, pilih Dasbor EC2.
4. Zona Ketersediaan tercantum di panel Kesehatan Layanan.

Untuk menemukan Availability Zone Anda menggunakan AWS CLI

- Gunakan perintah [describe-availability-zones](#) sebagai berikut untuk mendeskripsikan Zona Ketersediaan dalam Wilayah yang ditentukan yang diaktifkan untuk akun Anda.

```
aws ec2 describe-availability-zones --region region-name
```

- Gunakan perintah [describe-availability-zones](#) sebagai berikut untuk mendeskripsikan Zona Ketersediaan terlepas dari status keikutsertaan.

```
aws ec2 describe-availability-zones --all-availability-zones
```

Meluncurkan instans di Zona Ketersediaan

Saat Anda meluncurkan instans, pilih Wilayah yang menempatkan instans Anda lebih dekat dengan pelanggan tertentu, atau memenuhi persyaratan hukum atau lainnya yang Anda miliki. Dengan meluncurkan instans Anda di Zona Ketersediaan yang terpisah, Anda dapat melindungi aplikasi Anda dari kegagalan di satu lokasi.

Saat Anda meluncurkan sebuah instans, Anda dapat secara opsional menentukan Zona Ketersediaan di Wilayah yang Anda gunakan. Jika Anda tidak menentukan Zona Ketersediaan, kami memilih Zona Ketersediaan untuk Anda. Saat Anda meluncurkan instans awal, kami menyarankan Anda menerima Zona Ketersediaan default, karena ini memungkinkan kami memilih Zona Ketersediaan terbaik untuk Anda berdasarkan kesehatan sistem dan kapasitas yang tersedia. Jika

Anda meluncurkan instans tambahan, tentukan Zona Ketersediaan hanya jika instans baru Anda harus dekat dengan, atau dipisahkan dari, instans yang sedang berjalan.

Memigrasi sebuah instans ke Zona Ketersediaan lain

Jika perlu, Anda dapat memigrasi instans dari satu Zona Ketersediaan ke lainnya. Misalnya, jika Anda mencoba memodifikasi tipe instans dari instans Anda dan kami tidak dapat meluncurkan instans tipe baru di Zona Ketersediaan saat ini, Anda dapat memigrasikan instans ke Zona Ketersediaan yang memiliki kapasitas untuk tipe instans baru.

Proses migrasi meliputi:

- Membuat AMI dari instans asli
- Meluncurkan sebuah instans di Zona Ketersediaan yang baru
- Memperbarui konfigurasi instans baru, seperti yang ditunjukkan dalam prosedur berikut

Untuk memigrasi sebuah instans ke Zona Ketersediaan lain

1. Buat AMI dari instans. Prosedurnya tergantung pada jenis volume perangkat root untuk instance. Untuk informasi selengkapnya, lihat dokumentasi yang sesuai dengan volume perangkat root Anda:
 - [Buat AMI yang didukung Amazon EBS-Backed](#)
 - [Buat AMI Linux yang didukung penyimpanan instans](#)
2. Jika Anda perlu menjaga alamat IPv4 privat instans, Anda harus menghapus subnet di Zona Ketersediaan saat ini dan kemudian membuat subnet di Zona Ketersediaan baru dengan rentang alamat IPv4 yang sama dengan subnet asli. Perhatikan bahwa Anda harus menghentikan semua instans di subnet sebelum Anda dapat menghapusnya. Karena itu, Anda harus membuat AMI dari semua instans di subnet Anda sehingga Anda dapat memindahkan semua instans dari subnet saat ini ke subnet baru.
3. Luncurkan sebuah instans dari AMI yang baru saja Anda buat, dengan menentukan Zona Ketersediaan atau subnet baru. Anda dapat menggunakan tipe instans yang sama dengan instans asli, atau memilih tipe instans baru. Untuk informasi selengkapnya, lihat [Meluncurkan instans di Zona Ketersediaan](#).
4. Jika instans asli memiliki alamat IP Elastis terkait, kaitkan dengan instans baru. Untuk informasi selengkapnya, lihat [Pisahkan alamat IP Elastis](#).

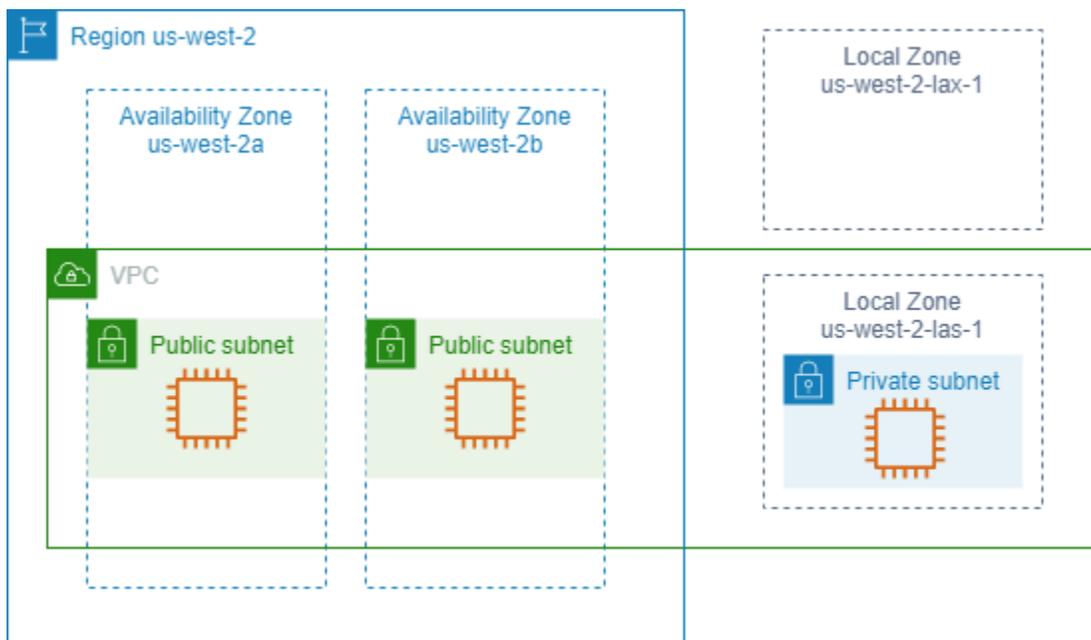
5. Jika instans asli adalah Instans Cadangan, ubah Zona Ketersediaan untuk reservasi Anda. (Jika Anda juga mengubah tipe instans, Anda juga dapat mengubah tipe instans untuk reservasi Anda.) Untuk informasi selengkapnya, lihat [Mengirimkan permintaan modifikasi](#).
6. (Opsional) Hentikan instans asli. Untuk informasi selengkapnya, lihat [Akhiri instans](#).

Zona Lokal

Zona Lokal adalah perpanjangan dari AWS Wilayah dalam kedekatan geografis dengan pengguna Anda. Local Zones memiliki koneksi sendiri ke internet dan dukungan AWS Direct Connect, sehingga sumber daya yang dibuat di Local Zone dapat melayani pengguna lokal dengan komunikasi latensi rendah. Untuk informasi selengkapnya, lihat [Apa itu AWS Local Zones?](#) di Panduan Pengguna AWS Local Zones.

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 dari Availability Zone-nya, dan dua Local Zone-nya. VPC mencakup Zona Ketersediaan dan salah satu Local Zones. Setiap zona di VPC memiliki satu subnet, dan setiap subnet memiliki instans.



Untuk menggunakan Local Zones, Anda harus mengaktifkannya terlebih dahulu. Untuk informasi selengkapnya, lihat [the section called “Menyertakan ke Local Zones”](#). Selanjutnya, buat subnet di Local Zones. Terakhir, luncurkan sumber daya di subnet Local Zones, seperti instans, sehingga aplikasi Anda dekat dengan pengguna Anda.

Daftar Isi

- [Local Zones yang Tersedia](#)
- [Menyertakan ke Local Zones](#)
- [Meluncurkan instans di Local Zones](#)

Local Zones yang Tersedia

Anda dapat menggunakan konsol Amazon EC2 atau antarmuka baris perintah untuk menentukan Local Zones mana yang tersedia untuk akun Anda. Untuk daftar lengkapnya, lihat [Lokasi AWS Local Zones](#).

Untuk menemukan Local Zones Anda menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari bilah navigasi, pilih pemilih Wilayah, kemudian pilih Wilayah induk.
3. Di panel navigasi, pilih Dasbor EC2.
4. Di pojok kanan atas halaman, pilih Atribut akun, Zona.

Untuk menemukan Local Zones Anda menggunakan AWS CLI

Gunakan perintah [describe-availability-zones](#) sebagai berikut untuk mendeskripsikan semua Local Zones di Wilayah yang ditentukan, meskipun tidak diaktifkan. Untuk mendeskripsikan hanya Local Zones yang telah Anda aktifkan, hilangkan `--all-availability-zones` opsi.

```
aws ec2 describe-availability-zones --region region-name --filters Name=zone-type,Values=local-zone --all-availability-zones
```

Menyertakan ke Local Zones

Sebelum Anda dapat menentukan Local Zones untuk sumber daya atau layanan, Anda harus ikut serta dalam Local Zones.

Pertimbangan

Beberapa AWS sumber daya mungkin tidak tersedia di semua Wilayah. Pastikan Anda dapat membuat sumber daya yang Anda perlukan di Wilayah atau Local Zones yang diinginkan sebelum meluncurkan sebuah instans di Local Zones tertentu. Untuk daftar layanan yang didukung di setiap Local Zones, lihat [Fitur AWS Local Zones](#).

Untuk menyertakan ke Local Zones Anda menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di sudut kiri atas halaman, pilih Pengalaman EC2 Baru. Anda tidak dapat menyelesaikan tugas ini menggunakan pengalaman konsol lama.
3. Dari bilah navigasi, pilih pemilih Wilayah, kemudian pilih Wilayah induk.
4. Di panel navigasi, pilih Dasbor EC2.
5. Di pojok kanan atas halaman, pilih Atribut akun, Zona.
6. Pilih Local Zone dan pilih Action > Manage Zone group.
7. Di bawah status Keikutsertaan, pilih Aktifkan.
8. Pilih Perbarui.

Untuk ikut serta dalam Local Zones menggunakan AWS CLI

Gunakan perintah [modify-availability-zone-group](#).

Meluncurkan instans di Local Zones

Saat Anda meluncurkan sebuah instans, Anda dapat menentukan subnet yang ada di Local Zones. Anda juga mengalokasikan alamat IP dari grup batas jaringan. Grup batas jaringan adalah serangkaian Zona Ketersediaan, Local Zones, atau Wavelength Zone yang unik, tempat AWS mengiklankan alamat IP, misalnya, `us-west-2-lax-1a`.

Anda dapat mengalokasikan alamat IP berikut dari grup batas jaringan:

- Alamat IPv4 Elastis yang disediakan Amazon
- Alamat VPC IPv6 yang disediakan Amazon (hanya tersedia di zona Los Angeles)

Untuk informasi selengkapnya tentang cara meluncurkan instance di Zona Lokal, lihat [Memulai AWS Local Zones](#) di Panduan Pengguna AWS Local Zones.

Wavelength Zones

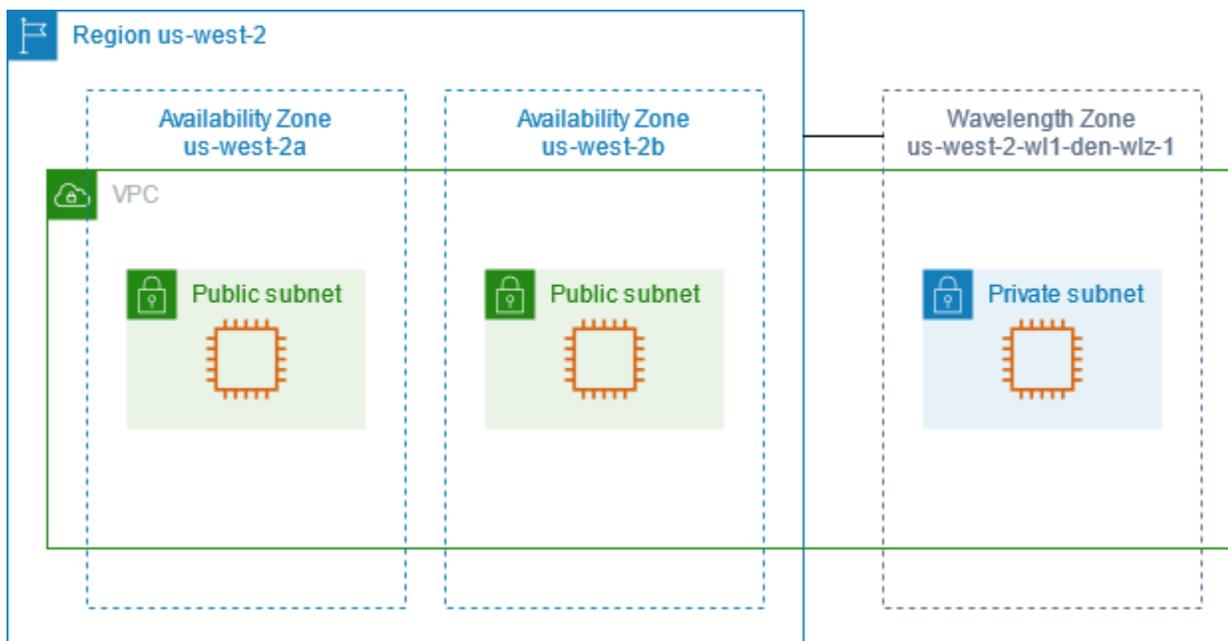
AWS Wavelength memungkinkan pengembang untuk membangun aplikasi yang memberikan latensi ultra-rendah ke perangkat seluler dan pengguna akhir. Wavelength menyebarkan layanan komputasi dan penyimpanan AWS standar ke tepi jaringan 5G operator telekomunikasi. Pengembang dapat

memperluas virtual private cloud (VPC) ke satu atau beberapa Wavelength Zone, dan kemudian menggunakan sumber daya seperti instans AWS Amazon EC2 untuk menjalankan aplikasi yang memerlukan latensi ultra-rendah dan koneksi ke layanan di Wilayah. AWS

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-w11-bos-w1z-1` di Boston.

Diagram berikut menggambarkan AWS Wilayah `us-west-2`, dua dari Availability Zone-nya, dan Wavelength Zone. VPC mencakup Zona Ketersediaan dan Wavelength Zone. Setiap zona di VPC memiliki satu subnet, dan setiap subnet memiliki instans.



Untuk menggunakan Wavelength Zone, Anda harus terlebih dahulu memilih Zona. Untuk informasi selengkapnya, lihat [the section called “Aktifkan Zona Panjang Gelombang”](#). Selanjutnya, membuat subnet di Wavelength Zone. Terakhir, luncurkan sumber daya Anda di subnet Wavelength Zone, sehingga aplikasi Anda lebih dekat dengan pengguna akhir.

Zona Panjang Gelombang tidak tersedia di setiap Wilayah. Untuk informasi tentang Wilayah yang mendukung Wavelength Zone, lihat [Wavelength Zone yang Tersedia](#) di Panduan Developer AWS Wavelength .

Daftar Isi

- [Menjelaskan Wavelength Zone Anda](#)
- [Aktifkan Zona Panjang Gelombang](#)
- [Meluncurkan instans dalam Wavelength Zone](#)

Menjelaskan Wavelength Zone Anda

Anda dapat menggunakan konsol Amazon EC2 atau antarmuka baris perintah untuk menentukan Zona Panjang Gelombang yang tersedia untuk akun Anda. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

Untuk menemukan Zona Panjang Gelombang Anda 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. Di panel navigasi, pilih Dasbor EC2.
4. Di pojok kanan atas halaman, pilih Atribut akun, Zona.

Untuk menemukan Zona Wavelength Anda menggunakan AWS CLI

- Gunakan perintah [describe-availability-zones](#) sebagai berikut untuk mendeskripsikan Wavelength Zone dalam Wilayah yang ditentukan yang diaktifkan untuk akun Anda.

```
aws ec2 describe-availability-zones --region region-name
```

- Gunakan perintah [describe-availability-zones](#) sebagai berikut untuk mendeskripsikan Wavelength Zone apa pun status keikutsertaannya.

```
aws ec2 describe-availability-zones --all-availability-zones
```

Aktifkan Zona Panjang Gelombang

Sebelum Anda menentukan Wavelength Zone untuk sumber daya atau layanan, Anda harus mengaktifkan Wavelength Zone.

Pertimbangan

- Beberapa AWS sumber daya tidak tersedia di semua Wilayah. Pastikan Anda dapat membuat sumber daya yang Anda perlukan di Wilayah atau Zona Panjang Gelombang yang diinginkan sebelum meluncurkan instans di Zona Panjang Gelombang tertentu.

Untuk memilih Zona Panjang Gelombang menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di sudut kiri atas halaman, pilih Pengalaman EC2 Baru. Anda tidak dapat menyelesaikan tugas ini menggunakan pengalaman konsol lama.
3. Dari bilah navigasi, pilih pemilih Wilayah, kemudian pilih Wilayah.
4. Di panel navigasi, pilih Dasbor EC2.
5. Di pojok kanan atas halaman, pilih Atribut akun, Zona.
6. Pilih Wavelength Zone dan pilih Action > Manage Zone group.
7. Di bawah status Keikutsertaan, pilih Aktifkan.
8. Pilih Perbarui.

Untuk mengaktifkan Wavelength Zones menggunakan AWS CLI

Gunakan perintah [modify-availability-zone-group](#).

Meluncurkan instans dalam Wavelength Zone

Saat Anda meluncurkan sebuah instans, Anda dapat menentukan subnet yang berada dalam Wavelength Zone. 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 informasi tentang cara meluncurkan instans di Wavelength Zone, lihat [Memulai AWS Wavelength](#) di Panduan Developer AWS Wavelength .

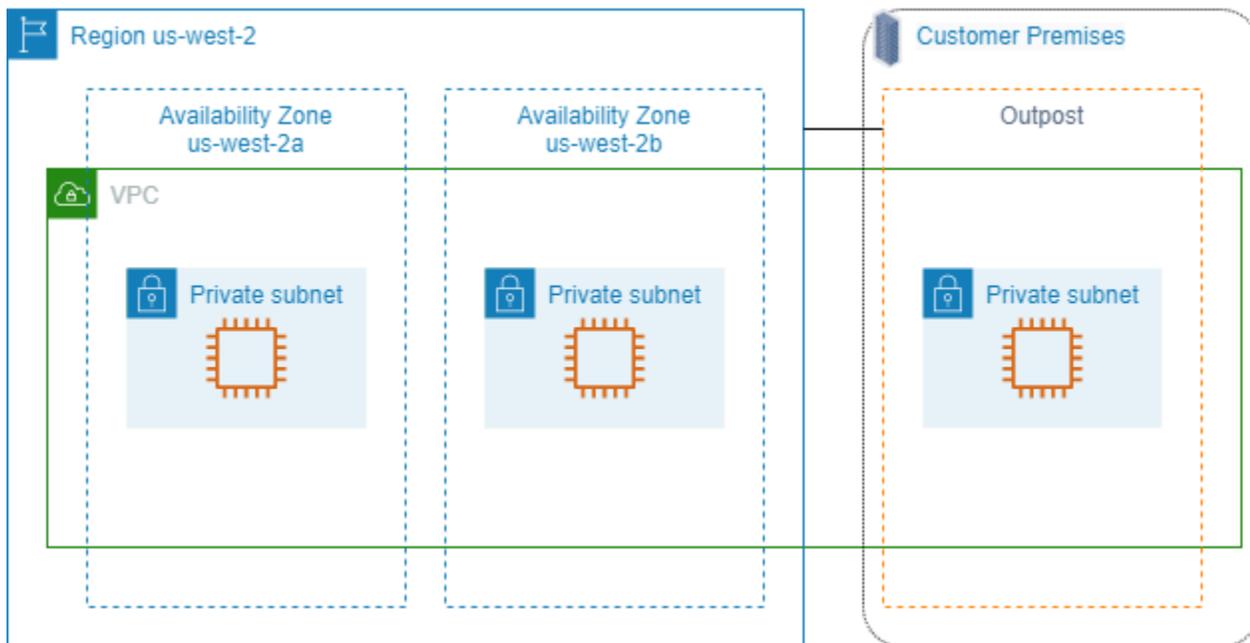
AWS Outposts

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 AWS Wilayah, 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 suatu AWS Wilayah. Anda dapat membuat subnet di Outpost Anda dan menentukannya saat Anda membuat AWS sumber daya. Instance dalam subnet Outpost berkomunikasi dengan instans lain di AWS Wilayah menggunakan alamat IP pribadi, semuanya dalam VPC yang sama.

Diagram berikut menggambarkan AWS Wilayah 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.



Untuk mulai menggunakan AWS Outposts, Anda harus membuat Outpost dan memesan kapasitas Outpost. Untuk informasi lebih lanjut tentang konfigurasi Outposts, lihat [katalog kami](#). Setelah peralatan Outpost diinstal, kapasitas komputasi dan penyimpanan tersedia untuk Anda saat meluncurkan instans Amazon EC2 di Outpost Anda.

Meluncurkan instans di Outpost

Anda dapat meluncurkan instans EC2 di subnet Outpost yang Anda buat. Grup keamanan mengontrol lalu lintas masuk dan ke luar untuk instans dengan antarmuka jaringan elastis di

subnet Outpost, seperti yang mereka lakukan untuk instans di subnet Zona Ketersediaan. Untuk menghubungkan ke instans EC2 di subnet Outpost, Anda dapat menentukan pasangan kunci saat Anda meluncurkan instans, seperti yang Anda lakukan untuk instans di subnet Zona Ketersediaan.

Kami menyarankan Anda membatasi volume root untuk sebuah instance pada 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 terpakai dari volume boot, lihat [Cara Membangun Volume EBS yang jarang di Blog Jaringan AWS Mitra](#).

Kami menyarankan Anda untuk meningkatkan waktu tunggu NVMe untuk volume root. Untuk informasi selengkapnya, lihat [batas waktu operasi I/O](#).

Untuk informasi tentang cara membuat Outpost, lihat [Memulai dengan AWS Outposts](#) di Panduan Pengguna AWS Outposts .

Buat volume di rak Outpost

AWS Outposts menawarkan faktor bentuk rak dan server. Jika kapasitas Anda ada 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](#).

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.

Daftar Isi

- [Alamat IPv4 privat](#)
- [Alamat IPv4 publik](#)
- [Optimalisasi alamat IPv4 publik](#)
- [Alamat IP elastis \(IPv4\)](#)
- [Alamat IPv6](#)
- [Bekerja dengan alamat IPv4 untuk instans Anda](#)
- [Bekerja dengan alamat IPv6 untuk instans Anda](#)
- [Beberapa alamat IP untuk instans EC2 Anda](#)
- [Konfigurasi alamat IPv4 privat sekunder untuk instans Windows Anda](#)
- [Nama host instans EC2](#)
- [Alamat link-lokal](#)

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 privat 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:

Subnet VPC dapat berupa salah satu tipe dari berikut ini:

- Subnet hanya IPv4: Anda hanya dapat membuat sumber daya di subnet ini dengan alamat IPv4 yang ditetapkan padanya.

- Subnet khusus IPv6: Anda hanya dapat membuat sumber daya di subnet ini dengan alamat IPv6 yang ditetapkan padanya.
- Subnet IPv4 dan IPv6: Anda dapat membuat sumber daya di subnet ini dengan alamat IPv4 atau IPv6 yang ditetapkan padanya.

Saat Anda meluncurkan instans EC2 ke subnet hanya IPv4 atau dual stack (IPv4 dan IPv6), instans menerima alamat IP privat primer 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 instans memiliki antarmuka jaringan default (eth0) yang diberi alamat IPv4 privat primer. 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 untuk instans EC2 Anda](#).

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:

- Memodifikasi atribut pengalamatan IP publik dari subnet Anda. Untuk informasi selengkapnya, lihat [Memodifikasi atribut pengalamatan IPv4 publik untuk subnet Anda](#) dalam Panduan Pengguna Amazon VPC.
- Mengaktifkan atau menonaktifkan fitur pengalamatan IP publik selama peluncuran, yang menggantikan atribut pengalamatan IP publik subnet. Untuk informasi selengkapnya, lihat [Menetapkan alamat IPv4 publik selama peluncuran instans](#).

- Anda dapat membatalkan penetapan alamat IP publik dari instans Anda setelah peluncuran dengan [mengelola alamat IP yang terkait dengan antarmuka jaringan](#).

Alamat IP publik ditetapkan ke instans Anda dari kumpulan alamat IPv4 publik Amazon, dan tidak terkait dengan akun Anda AWS . Ketika alamat IP publik tidak dikaitkan dengan instans Anda, alamat IP tersebut dilepas kembali ke kumpulan IPv4 publik, dan Anda tidak dapat menggunakannya kembali.

Dalam kasus tertentu, kami merilis alamat IP publik dari instans Anda, atau menyetapkannya yang baru:

- Kami melepas alamat IP publik instans Anda saat alamat IP dihentikan, dihibernasikan, atau dihentikan. Instans Anda yang dihentikan atau dihibernasikan menerima alamat IP publik baru saat dimulai.
- Kami melepas alamat IP publik instans Anda saat Anda mengaitkan alamat IP Elastis dengan instans tersebut. Ketika Anda memisahkan alamat IP Elastis dari instans Anda, maka instans tersebut menerima alamat IP publik yang baru.
- Jika alamat IP publik dari instans Anda di VPC telah dilepas, instans tersebut tidak akan menerima alamat IP yang baru jika ada lebih dari satu antarmuka jaringan yang disertakan ke instans Anda.
- Jika alamat IP publik instans Anda dilepas saat masih memiliki alamat IP privat sekunder yang dikaitkan dengan alamat IP Elastis, instans tidak menerima 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 IP elastis](#).

Note

- AWS mengenakan biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastis. 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.

Optimalisasi alamat IPv4 publik

AWS mengenakan biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastis. 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 [penyeimbang beban elastis](#) untuk memuat lalu lintas keseimbangan ke instans EC2 Anda dan [nonaktifkan Auto-assign IP publik pada ENI utama yang ditetapkan ke](#) instans. Load balancer 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 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 instans memiliki alamat IPv4 publik.
- Jika instans EC2 Anda berada di subnet publik dengan alamat IP publik yang dialokasikan untuk mereka, pertimbangkan untuk memindahkan instans ke subnet pribadi, menghapus alamat IP publik, dan menggunakan [gateway NAT](#) publik untuk memungkinkan 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 AWS Penagihan](#).

$$\text{NAT gateway per hour} + \text{NAT gateway public IPs} + \text{NAT gateway transfer} / \text{Existing public IP cost}$$

Di mana:

- $\text{NAT gateway per hour} = \$0.045 * 730 \text{ hours in a month} * \text{Number of Availability Zones the NAT gateways are in}$

- 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 publik milik Amazon.
- Matikan [auto-assign alamat IPv4 publik 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 dilampirkan ke instans Anda](#) telah menonaktifkan IP publik secara otomatis.
- [Konfigurasi 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 IP elastis (IPv4)

Alamat IP Elastis adalah alamat IPv4 publik yang dapat dialokasikan ke akun Anda. Anda dapat mengaitkannya ke dan memisahkannya dari instans sesuai kebutuhan. Alamat Elastic IP tersebut dialokasikan ke akun Anda sampai Anda memilih untuk melepaskannya. Untuk informasi selengkapnya tentang alamat IP Elastis dan cara menggunakannya, lihat [Alamat IP elastis](#).

Kami tidak mendukung alamat IP Elastis untuk IPv6.

Alamat IPv6

Atau, Anda dapat mengaitkan blok CIDR IPv6 dengan VPC Anda, dan mengaitkan blok CIDR IPv6 dengan subnet Anda. Blok CIDR IPv6 untuk VPC Anda secara otomatis ditetapkan dari

kumpulan alamat IPv6 Amazon; Anda tidak dapat memilih sendiri rentang tersebut. Untuk informasi selengkapnya, lihat topik berikut dalam Panduan Pengguna Amazon VPC:

- [Pengalamatan IP untuk VPC dan subnet Anda](#)
- [Tambahkan blok CIDR IPv6 ke VPC Anda](#)
- [Tambahkan blok CIDR IPv6 ke subnet Anda](#)

Alamat IPv6 bersifat unik secara global dan dapat dikonfigurasi agar tetap privat atau dapat dijangkau melalui Internet. Instans Anda menerima alamat IPv6 jika blok CIDR IPv6 dikaitkan dengan VPC dan subnet Anda, dan jika salah satu dari pernyataan berikut adalah benar:

- Subnet Anda dikonfigurasi untuk secara otomatis menetapkan alamat IPv6 ke sebuah instans selama peluncuran. Untuk informasi selengkapnya, lihat [Memodifikasi atribut pengalamatan IPv6 untuk 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.

Saat instans Anda menerima alamat IPv6 selama peluncuran, alamat tersebut dikaitkan dengan antarmuka jaringan primer (eth0) dari instans. Anda dapat mengelola alamat IPv6 untuk antarmuka jaringan primer instans Anda (eth0) dengan cara 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 per antarmuka jaringan per tipe instans](#).
- 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](#) atau [Mengelola alamat IP](#).

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 penempatannya terlebih dahulu.

Anda dapat mengontrol apakah instans dapat dicapai melalui alamat IPv6 mereka dengan mengontrol perutean untuk subnet Anda, 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 disimpan, lihat [Daftar Alamat Tujuan Khusus IANA IPv6](#) dan [RFC4291](#).

Bekerja dengan alamat IPv4 untuk instans 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.

Daftar Isi

- [Melihat alamat IPv4](#)
- [Menetapkan alamat IPv4 publik selama peluncuran instans](#)

Melihat alamat IPv4

Anda dapat menggunakan konsol Amazon EC2 untuk melihat alamat IPv4 publik dan privat dari instans Anda. Anda juga dapat menentukan alamat IPv4 publik dan alamat IPv4 privat dari instans Anda dari dalam instans Anda dengan menggunakan metadata instans. Untuk informasi selengkapnya, lihat [Cara menggunakan metadata instans](#).

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. Untuk menentukan alamat IPv4 publik instans Anda dari sebuah instans, gunakan metadata instans.

Untuk melihat alamat IPv4 instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).

Untuk menentukan alamat IPv4 instans Anda menggunakan metadata instans

1. Connect ke instans Anda. Untuk informasi selengkapnya, lihat [Connect ke instans EC2](#).
2. Gunakan perintah berikut untuk mengakses alamat IP pribadi.

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/local-ipv4
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4
```

Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/local-ipv4
```

3. Gunakan perintah berikut untuk mengakses alamat IP publik. Jika alamat IP Elastis dikaitkan dengan instans, nilai yang dikembalikan adalah alamat IP Elastis.

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/public-ipv4
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-ipv4
```

Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/public-ipv4
```

Menetapkan alamat IPv4 publik selama peluncuran instans

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 eth0. Fitur ini bergantung pada kondisi tertentu pada saat Anda meluncurkan instans Anda.

Pertimbangan

- Anda dapat membatalkan penetapan alamat IP publik dari instans Anda setelah peluncuran 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 pengaturan subnet menggunakan fitur tetapkan IP publik secara otomatis jika Anda menentukan antarmuka jaringan yang ada untuk eth0.
- Apakah Anda menetapkan alamat IP publik ke instans Anda selama peluncuran atau tidak, Anda dapat mengaitkan alamat IP Elastis dengan instans Anda setelah diluncurkan. Untuk informasi selengkapnya, lihat [Alamat IP elastis](#). Anda juga dapat mengubah perilaku pengalamatan IPv4 publik subnet Anda. Untuk informasi selengkapnya, lihat [Memodifikasi atribut pengalamatan IPv4 publik untuk subnet Anda](#).

Untuk menetapkan alamat IPv4 publik selama peluncuran instans menggunakan konsol

Ikuti prosedur untuk [meluncurkan instans](#), dan saat Anda mengonfigurasi [Pengaturan Jaringan](#), pilih opsi untuk menetapkan IP Publik secara otomatis.

Untuk mengaktifkan atau menonaktifkan fitur pengalamatan IP publik menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- Gunakan opsi `--associate-public-ip-address` atau `--no-associate-public-ip-address` dengan perintah [run-instances](#) (AWS CLI)

- Gunakan `-AssociatePublicIp` parameter dengan [New-EC2Instance](#) perintah (AWS Tools for Windows PowerShell)

Bekerja dengan alamat IPv6 untuk instans Anda

Anda dapat melihat alamat IPv6 yang ditetapkan untuk instans Anda, menetapkan alamat IPv6 publik untuk instans Anda, atau membatalkan penetapan alamat IPv6 dari instans Anda. Anda dapat melihat alamat ini di konsol melalui halaman Instans atau halaman Antarmuka Jaringan.

Daftar Isi

- [Melihat alamat IPv6](#)
- [Menetapkan alamat IPv6 ke sebuah instans](#)
- [Untuk membatalkan penetapan alamat IPv6 dari sebuah instans](#)

Melihat alamat IPv6

Anda dapat menggunakan konsol Amazon EC2 AWS CLI, dan metadata instans untuk melihat alamat IPv6 untuk instans Anda.

Untuk melihat alamat IPv6 untuk sebuah instans menggunakan konsol

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.

Untuk melihat alamat IPv6 instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).

Untuk melihat alamat IPv6 untuk sebuah instans menggunakan metadata instans

1. Connect ke instans Anda. Untuk informasi selengkapnya, lihat [Connect ke instans EC2](#).
2. Dapatkan alamat MAC dari instance dari `http://169.254.169.254/latest/meta-data/network/interfaces/macs/`.
3. Gunakan perintah berikut untuk melihat alamat IPv6.

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/network/interfaces/macs/mac-address/ipv6s
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

Menetapkan alamat IPv6 ke sebuah instans

Jika VPC dan subnet Anda memiliki blok CIDR IPv6 yang dikaitkan dengannya, Anda dapat menetapkan alamat IPv6 ke instans Anda selama atau setelah peluncuran. Alamat IPv6 ditetapkan dari rentang alamat IPv6 subnet, dan ditetapkan ke antarmuka jaringan dengan indeks perangkat `eth0`.

Untuk menetapkan alamat IPv6 selama peluncuran instans

Ikuti prosedur untuk [meluncurkan instans](#), dan saat Anda mengonfigurasi [Pengaturan Jaringan](#), pilih opsi untuk menetapkan IP IPv6 secara otomatis.

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. Masukkan alamat IPv6 dari rentang subnet atau biarkan kolom kosong agar Amazon dapat memilih alamat IPv6 untuk Anda.
5. Pilih Simpan.

Untuk menetapkan alamat IPv6 menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- Gunakan opsi `--ipv6-addresses` dengan perintah [run-instances](#) (AWS CLI)
- Gunakan `Ipv6Addresses` properti untuk `-NetworkInterface` dalam [New-EC2Instance](#) perintah (AWS Tools for Windows PowerShell)
- [assign-ipv6-addresses](#) (AWS CLI)
- [Register-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

Anda dapat membatalkan penetapan alamat IPv6 dari instans kapan saja.

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans menggunakan konsol

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.

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [unassign-ipv6-addresses](#) (AWS CLI)
- [Unregister-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell).

Beberapa alamat IP untuk instans EC2 Anda

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 per antarmuka jaringan per tipe instans](#).

Melakukan hal-hal berikut dapat bermanfaat saat Anda menetapkan beberapa alamat IP ke sebuah instans di VPC Anda:

- 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.

Daftar Isi

- [Cara kerja beberapa alamat IP](#)
- [Bekerja dengan beberapa alamat IPv4](#)
- [Bekerja dengan beberapa alamat IPv6](#)

Cara kerja beberapa alamat IP

Daftar berikut menjelaskan bagaimana beberapa alamat IP bekerja dengan antarmuka jaringan:

- 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.

Daftar berikut menjelaskan cara kerja beberapa alamat IP dengan alamat IP Elastis (hanya IPv4):

- Setiap alamat IPv4 privat dapat dikaitkan dengan satu alamat IP Elastis, begitu pula sebaliknya.
- Saat alamat IPv4 privat sekunder ditetapkan ulang ke antarmuka lain, alamat IPv4 privat sekunder mempertahankan keterkaitannya dengan alamat IP Elastis.
- Ketika alamat IPv4 privat sekunder tidak ditetapkan dari antarmuka, alamat IP Elastis terkait secara otomatis dipisahkan dari alamat IPv4 privat sekunder.

Bekerja dengan beberapa alamat IPv4

Anda dapat menetapkan alamat IPv4 privat sekunder ke sebuah instans, mengaitkan alamat Elastis IPv4 dengan alamat IPv4 privat sekunder, dan membatalkan penetapan alamat IPv4 privat sekunder.

Tugas

- [Menetapkan alamat IPv4 privat sekunder](#)
- [Konfigurasi sistem operasi untuk mengenali alamat IPv4 pribadi sekunder](#)
- [Melakukan Associate alamat IP Elastis dengan alamat IPv4 privat sekunder](#)

- [Melihat alamat IPv4 privat sekunder Anda](#)
- [Membatalkan penetapan alamat IPv4 privat sekunder](#)

Menetapkan alamat IPv4 privat sekunder

Anda dapat menetapkan alamat IPv4 privat sekunder ke antarmuka jaringan untuk sebuah instans saat Anda meluncurkan instans, atau setelah instans berjalan.

Untuk menetapkan alamat IPv4 privat sekunder saat meluncurkan sebuah instans

1. Ikuti prosedur untuk [meluncurkan instans](#). Untuk [pengaturan Jaringan](#), pilih Edit.
2. Pilih VPC dan subnet.
3. Perluas Konfigurasi jaringan lanjutan.
4. Untuk IP Sekunder, pilih Tetapkan secara otomatis dan masukkan jumlah alamat IP (Amazon secara otomatis menetapkan alamat IPv4 sekunder) atau pilih Tetapkan secara manual dan masukkan alamat IPv4.
5. Selesaikan langkah-langkah selanjutnya untuk [meluncurkan instans](#).

Untuk menetapkan alamat IPv4 sekunder selama peluncuran menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- Opsi `--secondary-private-ip-addresses` dengan perintah [run-instances](#) (AWS CLI)
- Tentukan `-NetworkInterface` dan tentukan `PrivateIpAddresses` parameter dengan [New-EC2Instance](#) perintah (AWS Tools for Windows PowerShell).

Untuk menetapkan alamat IPv4 privat sekunder ke antarmuka jaringan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Antarmuka Jaringan, lalu pilih antarmuka jaringan untuk instance.
3. Pilih Tindakan, Kelola Alamat IP.
4. Bentangkan antarmuka jaringan. Di bawah alamat IPv4, pilih Tetapkan alamat IP baru.
5. Masukkan alamat IPv4 tertentu yang berada dalam rentang subnet untuk instance, atau biarkan bidang kosong agar Amazon memilih alamat IPv4 untuk Anda.

6. (Opsional) Pilih Izinkan untuk mengizinkan alamat IP pribadi sekunder dipindahkan jika sudah ditetapkan ke antarmuka jaringan lain.
7. Pilih Simpan.

Atau, Anda dapat menetapkan alamat IPv4 privat sekunder ke sebuah instans. Pilih Instans di panel navigasi, pilih instans, lalu pilih Tindakan, Jaringan, Kelola Alamat IP. Anda dapat mengonfigurasi informasi yang sama seperti yang Anda lakukan pada langkah-langkah di atas. Alamat IP ditetapkan ke antarmuka jaringan primer (eth0) untuk instans.

Untuk menetapkan alamat IPv4 pribadi sekunder ke instance yang ada menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [assign-private-ip-addresses](#) (AWS CLI)
- [Register-EC2PrivateIpAddress](#) (AWS Tools for Windows PowerShell)

Konfigurasi sistem operasi untuk mengenali alamat IPv4 pribadi sekunder

Setelah Anda menetapkan alamat IPv4 privat sekunder ke instans, Anda perlu mengonfigurasi sistem operasi pada instans Anda untuk mengenali alamat IP privat sekunder.

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, mereshuffle alamat IPv4 sekunder selama pembaruan DHCP lease, dan memperbarui aturan perutean terkait. Anda dapat segera menyegarkan daftar antarmuka dengan menggunakan perintah `sudo service network restart` dan kemudian melihat up-to-date daftar menggunakan `ip addr li`. Jika Anda memerlukan kontrol manual atas konfigurasi jaringan Anda, Anda dapat menghapus paket `ec2-net-utils`. Untuk informasi selengkapnya, lihat [Mengonfigurasi antarmuka jaringan Anda menggunakan ec2-net-utils untuk Amazon Linux 2](#).
- 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 [Konfigurasi alamat IPv4 privat sekunder untuk instans Windows Anda](#).

Melakukan Associate alamat IP Elastis dengan alamat IPv4 privat sekunder

Untuk mengaitkan alamat IP Elastis dengan alamat IPv4 privat sekunder

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih kotak centang untuk alamat IP Elastis
4. Pilih Actions, Associate Elastic IP Address.
5. Untuk jenis Sumber Daya, pilih Antarmuka jaringan. pilih antarmuka jaringan, lalu pilih alamat IP sekunder dari daftar alamat IP Pribadi.
6. Untuk antarmuka Jaringan, pilih antarmuka jaringan. pilih alamat IP sekunder dari daftar alamat IP pribadi.
7. Untuk alamat IP pribadi, pilih alamat IP sekunder.
8. Pilih Kaitkan.

Untuk mengaitkan alamat IP Elastis dengan alamat IPv4 privat sekunder menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [associate-address](#) (AWS CLI)
- [Register-EC2Address](#) (AWS Tools for Windows PowerShell)

Melihat alamat IPv4 privat sekunder Anda

Untuk melihat alamat IPv4 privat yang ditetapkan ke 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. Pada tab Detail, di bawah alamat IP, cari alamat IPv4 pribadi dan alamat IPv4 pribadi sekunder.

Untuk melihat alamat IPv4 privat yang ditetapkan ke instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih kotak centang untuk contoh.
4. Pada tab Jaringan, di bawah Detail jaringan, cari alamat IPv4 pribadi dan alamat IPv4 pribadi sekunder.

Membatalkan penetapan alamat IPv4 privat sekunder

Jika Anda tidak lagi memerlukan alamat IPv4 privat sekunder, Anda dapat membatalkan penetapannya dari instans atau antarmuka jaringan. Ketika alamat IPv4 privat sekunder tidak ditetapkan dari antarmuka jaringan, alamat IP Elastis (jika ada) juga akan dipisahkan.

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 Unassign untuk alamat IPv4 untuk membatalkan penetapan.
5. Pilih Simpan.

Untuk membatalkan penetapan alamat IPv4 privat sekunder 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, pilih Tindakan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Untuk alamat IPv4, pilih Unassign untuk alamat IPv4 untuk membatalkan penetapan.
5. Pilih Simpan.

Untuk membatalkan penetapan alamat IPv4 privat sekunder menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [unassign-private-ip-addresses](#) (AWS CLI)
- [Unregister-EC2PrivateIpAddress](#) (AWS Tools for Windows PowerShell)

Bekerja dengan beberapa alamat IPv6

Anda dapat menetapkan beberapa alamat IPv6 ke instans Anda, melihat alamat IPv6 yang ditetapkan ke instans Anda, dan membatalkan penetapan alamat IPv6 dari instans Anda.

Daftar Isi

- [Menetapkan beberapa alamat IPv6](#)
- [Melihat alamat IPv6 Anda](#)
- [Batalkan penetapan alamat IPv6](#)

Menetapkan beberapa alamat IPv6

Anda dapat menetapkan satu atau beberapa alamat IPv6 ke instans Anda selama peluncuran atau setelah peluncuran. Untuk menetapkan alamat IPv6 ke sebuah instans, VPC dan subnet tempat Anda meluncurkan instans harus memiliki blok CIDR IPv6 terkait.

Untuk menetapkan beberapa alamat IPv6 selama peluncuran

1. Ikuti prosedur untuk [meluncurkan instans](#). Untuk [pengaturan Jaringan](#), pilih Edit.
2. Pilih VPC dan subnet.
3. Perluas Konfigurasi jaringan lanjutan.
4. Untuk IP IPv6, pilih Tetapkan secara otomatis dan jumlah alamat IP (Amazon secara otomatis menetapkan alamat IPv6) atau pilih Tetapkan secara manual dan masukkan alamat IPv6.
5. Selesaikan langkah-langkah selanjutnya untuk [meluncurkan instans](#).

Anda dapat menggunakan layar Instans konsol Amazon EC2 untuk menetapkan beberapa alamat IPv6 ke instans yang ada. Ini menetapkan alamat IPv6 ke antarmuka jaringan primer (eth0) untuk instans. Untuk menetapkan alamat IPv6 tertentu ke instans, pastikan bahwa alamat IPv6 belum ditetapkan ke instans atau antarmuka jaringan lain.

Untuk menetapkan beberapa alamat IPv6 ke instans yang ada

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda, pilih Tindakan, Jaringan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Untuk alamat IPv6, pilih Tetapkan alamat IP baru untuk setiap alamat IPv6 yang akan ditambahkan. Anda dapat menentukan alamat IPv6 dari rentang subnet, atau membiarkan bidang kosong untuk membiarkan Amazon memilih alamat IPv6 untuk Anda.
5. Pilih Simpan.

Atau, Anda dapat menetapkan beberapa IPv6 ke antarmuka jaringan yang sudah ada. Antarmuka jaringan harus dibuat di subnet yang memiliki blok CIDR IPv6 terkait. Untuk menetapkan alamat IPv6 tertentu ke antarmuka jaringan, pastikan bahwa alamat IPv6 belum ditetapkan ke antarmuka jaringan lain.

Untuk menetapkan beberapa alamat IPv6 ke 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 Anda, pilih Tindakan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Untuk alamat IPv6, pilih Tetapkan alamat IP baru untuk setiap alamat IPv6 yang akan ditambahkan. Anda dapat menentukan alamat IPv6 dari rentang subnet, atau membiarkan bidang kosong untuk membiarkan Amazon memilih alamat IPv6 untuk Anda.
5. Pilih Simpan.

Ikhtisar CLI

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- Menetapkan alamat IPv6 selama peluncuran:
 - Gunakan opsi `--ipv6-addresses` atau `--ipv6-address-count` dengan perintah [run-instances](#) (AWS CLI)
 - Tentukan `-NetworkInterface` dan tentukan `Ipv6AddressCount` parameter `Ipv6Addresses` atau dengan [New-EC2Instance](#) perintah (AWS Tools for Windows PowerShell).
- Menetapkan alamat IPv6 ke antarmuka jaringan:
 - [assign-ipv6-addresses](#) (AWS CLI)
 - [Register-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

Melihat alamat IPv6 Anda

Anda dapat melihat alamat IPv6 untuk instans atau untuk antarmuka jaringan.

Untuk melihat alamat IPv6 yang ditetapkan ke instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih kotak centang untuk contoh Anda.
4. Pada tab Jaringan, cari bidang alamat IPv6.

Untuk melihat alamat IPv6 yang ditetapkan ke 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 Anda.
4. Pada tab Detail, di bawah alamat IP, cari bidang alamat IPv6.

Ikhtisar CLI

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- Melihat alamat IPv6 untuk sebuah instans:
 - [describe-instances](#) (AWS CLI)
 - [Get-EC2Instance](#) (AWS Tools for Windows PowerShell).
- Melihat alamat IPv6 untuk antarmuka jaringan:
 - [describe-network-interfaces](#) (AWS CLI)
 - [Get-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Batalkan penetapan alamat IPv6

Anda dapat membatalkan penetapan alamat IPv6 dari antarmuka jaringan primer sebuah instans, atau Anda dapat membatalkan penetapan alamat IPv6 dari antarmuka jaringan.

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 kotak centang untuk instans Anda, lalu pilih Tindakan, Jaringan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Di bawah alamat IPv6, pilih Batalkan penetapan di samping alamat IPv6.
5. Pilih Simpan.

Untuk membatalkan penetapan alamat IPv6 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 Anda, lalu pilih Tindakan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Di bawah alamat IPv6, pilih Batalkan penetapan di samping alamat IPv6.
5. Pilih Simpan.

Ikhtisar CLI

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [unassign-ipv6-addresses](#) (AWS CLI)
- [Unregister-EC2Ipv6 AddressList](#) (AWS Tools for Windows PowerShell)

Konfigurasi alamat IPv4 privat sekunder untuk instans Windows Anda

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: Konfigurasi alamat IP statis dalam instans Anda](#)
- [Langkah 2: Konfigurasi alamat IP privat sekunder untuk instans Anda](#)
- [Langkah 3: Konfigurasi aplikasi untuk Menggunakan alamat IP privat sekunder](#)

Prasyarat

1. 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 IPv4 privat sekunder](#).
2. Alokasikan alamat IP Elastis dan kaitkan dengan alamat IPv4 privat sekunder. Untuk informasi lebih lanjut, lihat [Mengalokasikan alamat IP Elastis](#) dan [Melakukan Associate alamat IP Elastis dengan alamat IPv4 privat sekunder](#)

Langkah 1: Konfigurasi alamat IP statis dalam 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 dalam instans Anda, alamat IP harus sama persis 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 IPv4 Address, Subnet Mask, Default Gateway, dan DNS Server 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 Tcpi. . . . . : Enabled
```

3. Buka Network and Sharing Center dengan menjalankan perintah berikut di PowerShell:

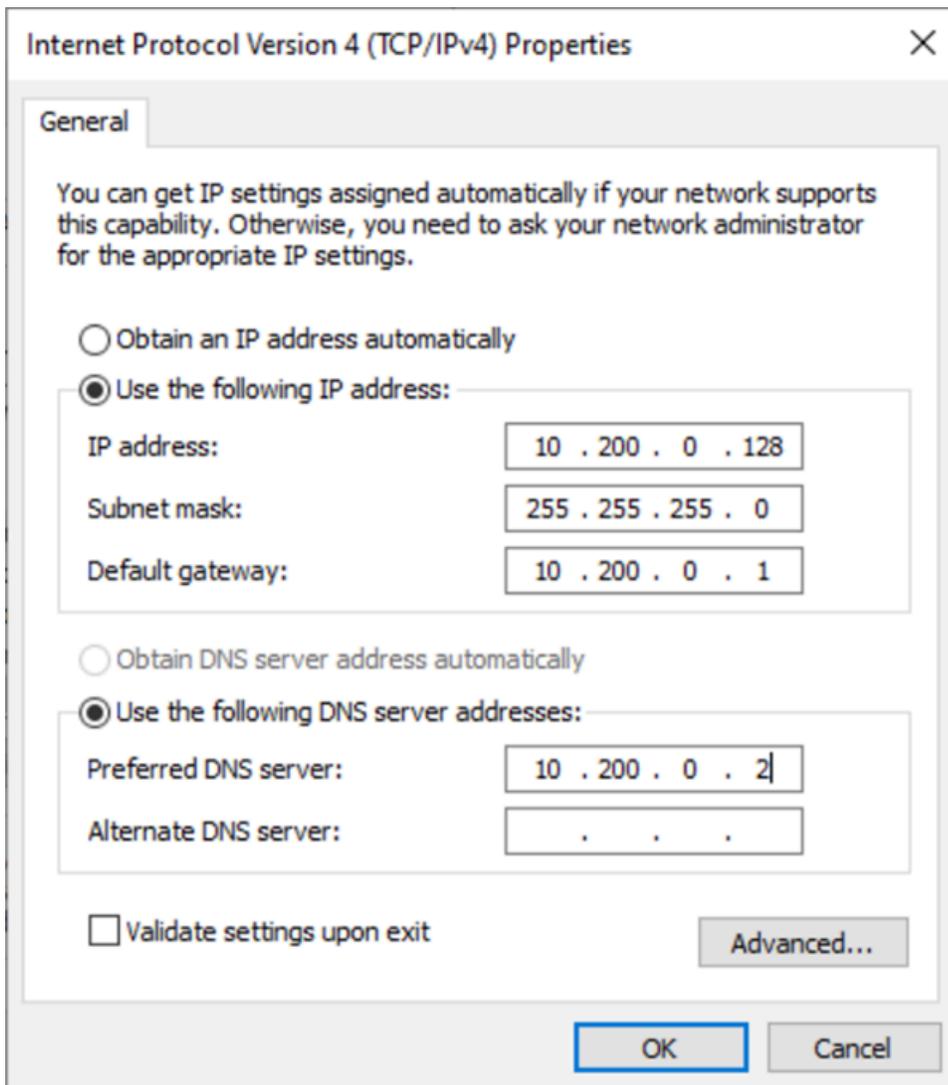
```
& $env:SystemRoot\system32\control.exe ncpa.cpl
```

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), Properti.
6. Di kotak dialog Properti Protokol Internet Versi 4 (TCP/IPv4), pilih Gunakan alamat IP berikut, masukkan nilai berikut, kemudian 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.



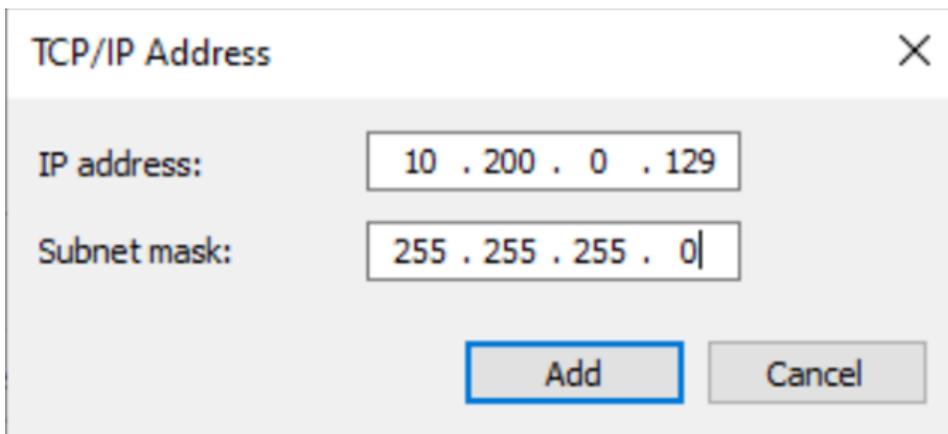
Anda akan kehilangan konektivitas RDP ke instans Windows selama beberapa detik saat instans tersebut diubah dari menggunakan DHCP menjadi pengalamatan statis. Instans tersebut mempertahankan informasi alamat IP yang sama seperti sebelumnya, tetapi sekarang informasi ini statis dan tidak dikelola oleh DHCP.

Langkah 2: Konfigurasi alamat IP privat sekunder untuk instans Anda

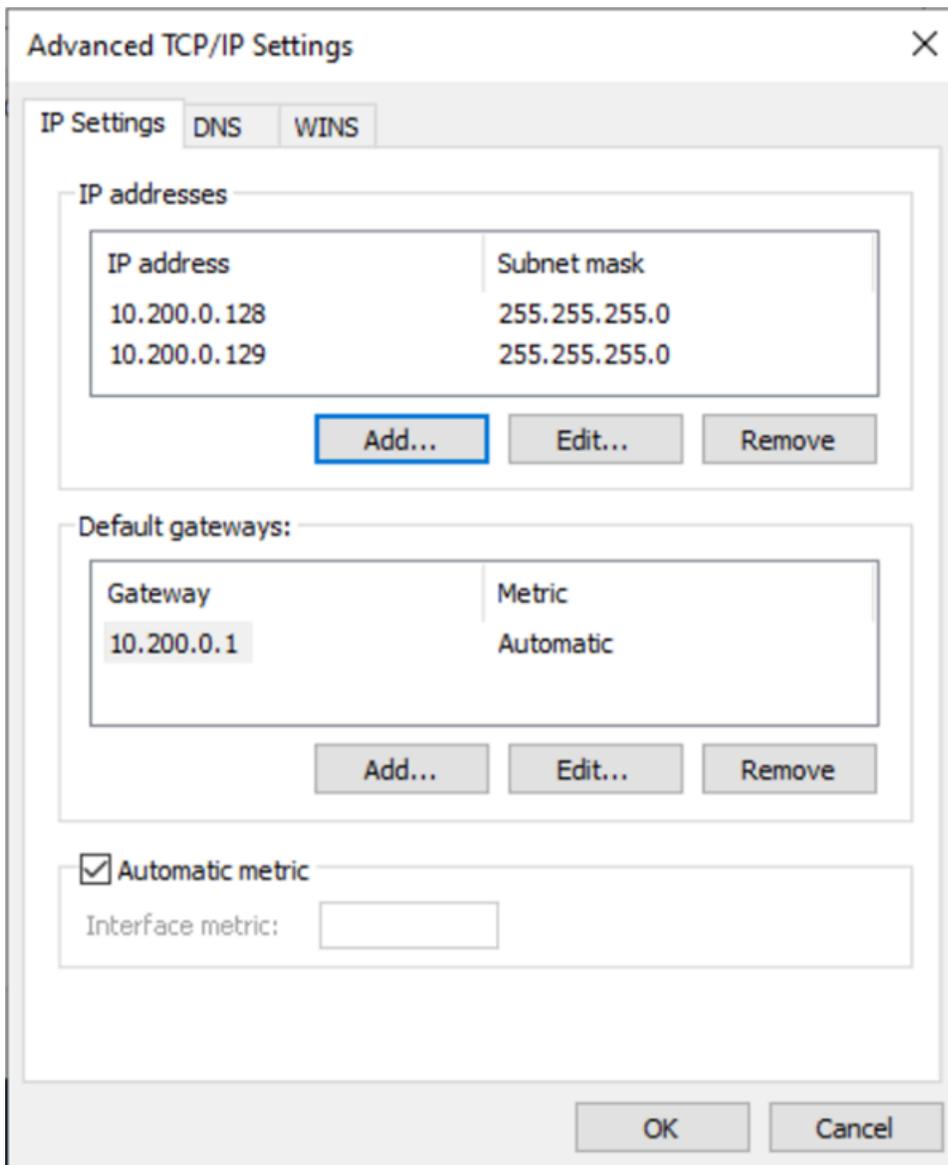
Setelah menyiapkan pengalamatan IP statis pada instans Windows, Anda siap untuk menyiapkan alamat IP privat kedua.

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 Properti Koneksi Area Lokal, pilih Protokol Internet Versi 4 (TCP/IPv4), Properti, Lanjutan.
9. Pilih Tambahkan.
10. Di kotak dialog Alamat TCP/IP, ketik alamat IP pribadi sekunder untuk alamat IP. Untuk Subnet mask, ketik subnet mask yang sama dengan yang Anda masukkan untuk alamat IP privat primer [Langkah 1: Konfigurasi alamat IP statis dalam 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)

```

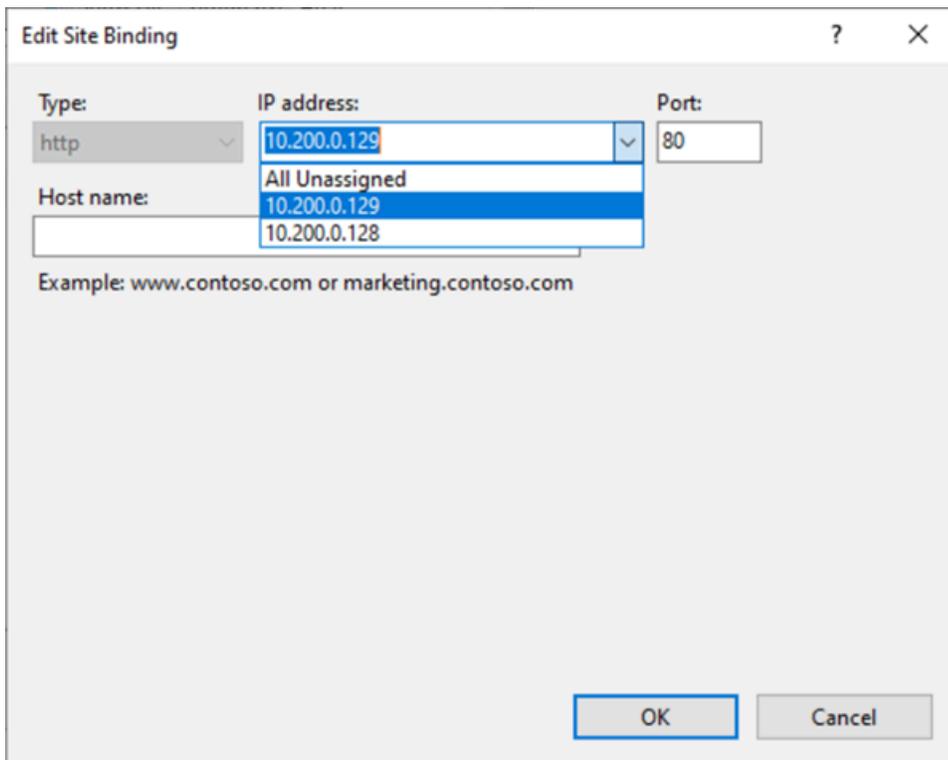
```
Subnet Mask . . . . . : 255.255.255.0
IPv4 Address. . . . . : 10.200.0.129(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.200.0.1
DHCPv6 IAID . . . . . : 151166011
DHCPv6 Client DUID. . . . . : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-E7
DNS Servers . . . . . : 10.200.0.2
NetBIOS over Tcpi. . . . . : Enabled
```

Langkah 3: Konfigurasi 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.)



The screenshot shows the 'Edit Site Binding' dialog box. It has three main sections: 'Type' with a dropdown menu set to 'http'; 'IP address' with a text box containing '10.200.0.129'; and 'Port' with a text box containing '80'. Below these is a 'Host name' section with a dropdown menu that is open, showing 'All Unassigned', '10.200.0.129', and '10.200.0.128'. Underneath the dropdown is an example: 'Example: www.contoso.com or marketing.contoso.com'. At the bottom right are 'OK' and 'Cancel' buttons.

7. Pilih OK, Tutup.

Nama host instans EC2

Saat Anda membuat instance EC2, buat nama host untuk AWS instance itu. Untuk informasi selengkapnya tentang jenis nama host dan cara AWS menyediakannya, lihat [Tipe nama host instans Amazon EC2](#) Amazon menyediakan server DNS yang menyelesaikan nama host yang disediakan Amazon 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.

Alamat link-lokal

Alamat link-lokal adalah alamat IP yang terkenal dan tidak dapat dirutekan. Amazon EC2 menggunakan alamat dari ruang alamat link-local untuk menyediakan layanan yang hanya dapat diakses dari instans EC2. Layanan ini tidak berjalan pada instans, mereka berjalan di host yang mendasarinya. Saat Anda mengakses alamat link-lokal untuk layanan ini, Anda berkomunikasi dengan hypervisor Xen atau pengontrol Nitro.

Rentang alamat link-lokal

- IPv4 — 169.254.0.0/16 (169.254.0.0 sampai 169.254.255.255)
- IPv6 – fe80::/10

Layanan yang Anda akses menggunakan alamat link-lokal

- [Layanan Metadata Instans](#)
- [Amazon Route 53 Resolver](#) (juga dikenal sebagai server DNS Amazon)
- [Layanan Amazon Time Sync](#)

Tipe nama host instans Amazon EC2

Bagian ini menjelaskan tipe nama host OS tamu instans Amazon EC2 yang tersedia saat Anda meluncurkan instans ke subnet VPC Anda.

Nama host membedakan instans EC2 di jaringan Anda. Anda dapat menggunakan nama host dari sebuah instans jika, misalnya, Anda ingin menjalankan skrip untuk berkomunikasi dengan beberapa atau semua instans di jaringan Anda.

Daftar Isi

- [Tipe nama host EC2](#)
- [Di mana Anda melihat Nama sumber daya dan nama IP](#)
- [Cara memutuskan apakah akan memilih nama Sumber Daya atau nama IP](#)
- [Modifikasi tipe Nama Host dan konfigurasi Nama host DNS](#)

Tipe nama host EC2

Ada dua tipe nama host 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 Privat, nama ini dapat mengembalikan alamat IPv4 privat (Catatan A) dan/atau Alamat Unicast Global IPv6 (catatan AAAA).

Tipe nama host OS tamu instans EC2 bergantung pada pengaturan subnet:

- Jika instans diluncurkan ke subnet hanya IPv4, 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 instans diluncurkan ke subnet hanya IPv6, 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 AWS Wilayah 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 subnet hanya IPv6, Tipe nama host dari Nama sumber daya dipilih secara default. Saat Anda meluncurkan instans 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 [Modifikasi tipe Nama Host dan konfigurasi Nama host DNS](#).

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 AWS Wilayah 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 privat berdasarkan nama sumber daya, Anda dapat mengonfigurasi apakah kueri DNS A dan/atau DNS AAAA untuk instans ditanggapi atau tidak. Anda mengonfigurasi perilaku respons saat meluncurkan instans atau memodifikasi subnet. Untuk informasi selengkapnya, lihat [Modifikasi tipe Nama Host dan konfigurasi Nama host DNS](#).

Di mana Anda melihat Nama sumber daya dan nama IP

Bagian ini menjelaskan di mana Anda melihat nama sumber daya tipe nama host dan nama IP di konsol 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 instans EC2 di wizard (lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#)) dan, ketika Anda mengonfigurasi detailnya, Anda memilih subnet yang Anda konfigurasikan menjadi hanya IPv6.

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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#)) 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.

Cara memutuskan apakah akan memilih nama Sumber Daya atau nama IP

Saat Anda meluncurkan instans EC2 (lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#)), 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.

Modifikasi tipe Nama Host dan konfigurasi Nama host DNS

Ikuti langkah-langkah di bagian ini untuk memodifikasi tipe Nama host dan konfigurasi Nama host DNS untuk subnet atau instans EC2 setelah diluncurkan.

Daftar Isi

- [Subnet](#)
- [Instans EC2](#)

Subnet

Modifikasi konfigurasi untuk subnet dengan memilih subnet di konsol VPC dan memilih Tindakan, Edit pengaturan subnet.

Note

Mengubah pengaturan subnet tidak mengubah konfigurasi instans EC2 yang sudah diluncurkan di subnet.

- Tipe nama host: Menentukan apakah Anda ingin pengaturan default nama host OS tamu dari instans EC2 yang diluncurkan di subnet menjadi nama sumber daya atau nama IP.
- Aktifkan permintaan IPv4 nama host DNS (catatan A): Menentukan apakah permintaan/kueri DNS ke nama sumber daya Anda diselesaikan ke alamat IPv4 privat (catatan A) dari instans EC2 ini.
- Aktifkan permintaan nama host DNS IPv6 (catatan AAAA): Menentukan apakah permintaan/kueri DNS ke nama sumber daya Anda diselesaikan ke alamat IPv6 (catatan AAAA) dari instans EC2 ini.

Instans EC2

Ikuti langkah-langkah di bagian ini untuk memodifikasi tipe Nama host dan konfigurasi Nama Host DNS untuk instans EC2.

Important

- Untuk mengubah pengaturan Gunakan nama berbasis sumber daya sebagai nama host OS tamu, Anda harus menghentikan instans tersebut terlebih dahulu. Untuk mengubah pengaturan Menjawab permintaan nama host DNS IPv4 (catatan A) atau Menjawab permintaan nama host DNS IPv6 (catatan AAAA), Anda tidak perlu menghentikan instans.
- Untuk mengubah pengaturan apa pun untuk tipe instans EC2 yang tidak didukung EBS, Anda tidak dapat menghentikan instans. Anda harus menghentikan instans dan meluncurkan instans baru dengan tipe Nama host yang diinginkan dan konfigurasi Nama host DNS.

Untuk memodifikasi tipe Nama Host dan konfigurasi Nama host DNS untuk instans EC2

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.
 - Menjawab permintaan IPv4 nama host DNS (catatan A): Menentukan apakah permintaan/kueri DNS ke nama sumber daya Anda diselesaikan ke alamat IPv4 privat dari instans EC2 ini.
 - Menjawab permintaan nama host DNS IPv6 (catatan AAAA): Menentukan apakah permintaan/kueri DNS ke nama sumber daya Anda diselesaikan ke alamat IPv6 (catatan AAAA) dari instans EC2 ini.
4. Pilih Simpan.
5. Jika Anda menghentikan instans, mulai lagi.

Bring your own IP addresses (BYOIP) di Amazon EC2

Anda dapat membawa sebagian atau seluruh rentang alamat IPv4 atau IPv6 yang dapat dirutekan secara publik dari jaringan lokal ke akun Anda. AWS Anda terus mengontrol rentang alamat dan Anda dapat mengiklankan rentang alamat di internet melalui AWS. Setelah Anda membawa rentang alamat ke AWS, itu muncul di AWS akun Anda sebagai kumpulan alamat.

Untuk daftar Wilayah tempat BYOIP tersedia, lihat [Ketersediaan wilayah](#).

Note

- Langkah-langkah di halaman 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 [Membawa alamat IP Anda sendiri \(BYOIP\) di Panduan AWS Global Accelerator Pengembang](#).

- 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.

Daftar Isi

- [Definisi BYOIP](#)
- [Persyaratan dan kuota](#)
- [Prasyarat orientasi untuk rentang alamat BYOIP Anda](#)
- [Onboard BYOIP Anda](#)
- [Menggunakan rentang alamat Anda](#)
- [Validasi BYOIP Anda](#)
- [Ketersediaan wilayah](#)
- [Ketersediaan Local Zone](#)
- [Pelajari selengkapnya](#)

Definisi BYOIP

- Sertifikat X.509 Self-sign — Standar sertifikat yang paling umum digunakan untuk mengenkripsi dan mengautentikasi data dalam jaringan. Ini adalah sertifikat yang digunakan oleh AWS untuk memvalidasi kontrol atas ruang IP dari catatan RDAP. [Untuk informasi selengkapnya tentang sertifikat X.509, lihat RFC 3280.](#)
- 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”. Tipe catatan tertentu perlu diperbarui oleh pelanggan melalui mekanisme yang disediakan RIR. 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 terkait wilayah geografis. BYOIP saat ini mendukung pendaftaran di American Registry for Internet Numbers (ARIN), Réseaux IP Européens Network Coordination Centre (RIPE), atau Asia-Pacific Network Information Centre (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 /56 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 Wilayah ke akun Anda. [AWS Anda tidak dapat menyesuaikan kuota untuk CIDR BYOIP menggunakan konsol Service Quotas, tetapi Anda dapat meminta peningkatan kuota dengan menghubungi Pusat AWS Dukungan seperti yang dijelaskan dalam kuota layanan di AWSReferensi Umum AWS](#)
- Anda tidak dapat membagikan rentang alamat IP Anda dengan akun lain AWS RAM kecuali Anda menggunakan Amazon VPC IP Address Manager (IPAM) dan mengintegrasikan IPAM dengan Organizations. AWS Untuk informasi selengkapnya, lihat [Mengintegrasikan IPAM dengan AWS Organizations](#) 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 sehari-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 tunggal dan catatan.
- BYOIP tidak didukung untuk Wavelength Zones atau on. 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 AWS, Anda dapat menggunakan semua alamat IP dalam rentang tersebut, termasuk alamat pertama (alamat jaringan) dan alamat terakhir (alamat siaran).

Prasyarat orientasi untuk rentang alamat BYOIP Anda

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.

Fase persiapan

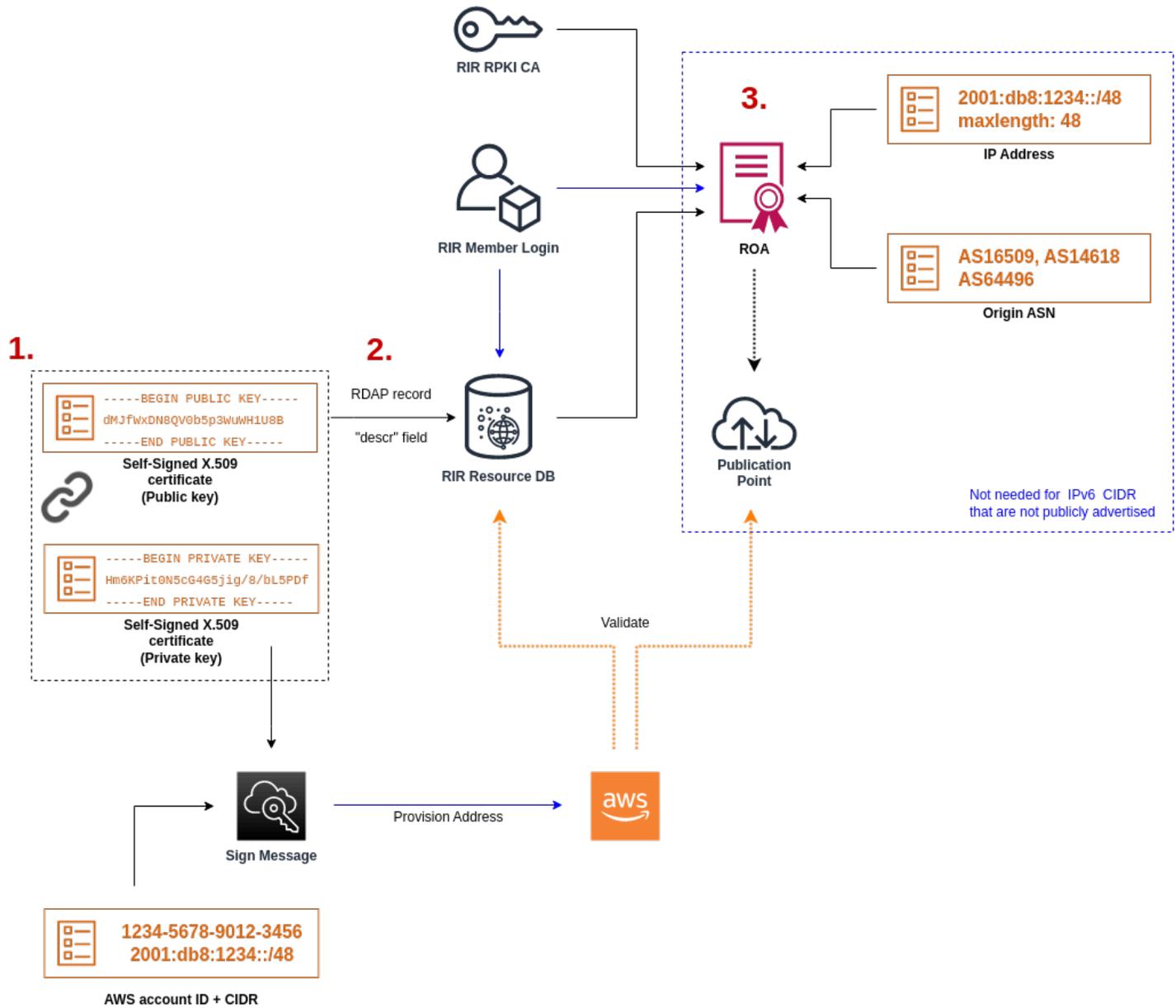
1. [Buat kunci privat](#) dan gunakan untuk membuat sertifikat X.509 yang ditandatangani sendiri untuk tujuan autentikasi. Sertifikat ini hanya digunakan selama fase penyediaan.

Fase konfigurasi RIR

2. [Mengunggah sertifikat yang ditandatangani sendiri](#) ke 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, persiapan dan langkah 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.

⚠ Important

Sebelum melakukan orientasi rentang alamat Anda, lengkapi prasyarat berikut. Tugas-tugas pada bagian ini memerlukan terminal Linux dan dapat dijalankan menggunakan Linux, [AWS CloudShell](#), atau [Windows Subsystem for Linux](#).

1. Buat kunci privat dan buat sertifikat X.509

Gunakan prosedur berikut untuk membuat sertifikat X.509 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: xxxxxxxx  
Verifying - Enter PEM pass phrase: xxxxxxxx
```

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: xxxxxxxx
```

```

-----BEGIN PRIVATE KEY-----
MIIEvGIBADANBqkqhkiG9w0BAQEFAASCbGwggSkAgEAAoIBAQDFBXHRI4HVKAhH
3seiciooizCRTbJe1+YsXNTja4XyKypVGIFWDGhZs44FCH1P00SVJ+NqP74w96oM
7DPS3xo9kaQyZBFn2YEp2EBq5vf307KHNRmZZUmkn0zH0SEpNmY2fMxISBxewLxR
FAniwmSd/8TDvHJMY9FvAIVWuTsv510tJKk+a91K4+t03UdDR7Sno5WEXefsBrW3
g1ydo3TBsx8i5/YiV0cNApy7ge2/FiwY3aCXJB6r6nuF6H8mRgI4r4vkMRs01AhJ
DnZPNNeweboo+K3Q31wbgbm0KD/z9svk8N/+hUTBtIX0fRtbG+PLIw3xWRHGfMSn2
BzsPVuDLAgMBAAEcggEACiJUj2hfJkKv47Dc3es3Zex67A5uDVjXmxf0x2Xhdupn
fAcNqAptV6fXt0SPUNbhUxbBKNbshoJGufFwXPLi1SxnpzvkdU4Hyco4zgbhXfSE
RNYjYf0GzTPwdBLpNMB6k3Tp4RHse6dNrLH0jDhpIoL8cQEBdBjyVF5X0wymEbmV
mC0jgH/MxsBAPWW6ZKicg9ULM1WiAZ3MRAZPjHHgpYkAAsUWKAbCBwVQcVjG059W
jfZjzTX5pQtVvH68rucih88DTZCwjCkjbHxg+0IkJBLE5wkh82jIHSivZ63flwLw
z+E0+HhELSZJrn2MY6Jxmik3qNNUOF/Z+3msdj2luQKBgQDjw1C/3jxp8zJy6P8o
JQKv7TdvMwUj4VSW0HZBHLv4evJaaia0uQjIo1UDa8AYitqhX1NmCCehGH8yuXj/
v6V3CzMKDkmRr1Nr0NnSz5QsndQ04Z6ihAQ1PmJ96g4wKtgoC7AYpyP0g1a+4/sj
b1+o3YQI4pD/F71c+qaztH7PRwKBgQDdc23yNmT3+Jyptf0fKjEv0NK+xwUKzi9c
L/0zBq5y0IC1Pz2T85g0e1i8kwZws+xlpG6uBT6lmIJELd0k59FyupNu4dPvX5SD
6GGqd4xjk9KvI74usGe0BohmF0phTHkrWKBxXiyT0oS8zjnJlEn8ysIpGg028jJr
LpaHNZ/MXQKBgQDFLncnS0LzpsS2aK0tzyZU8SMYqVH0GMxj7quhneBq2T6FbiLD
T9TV1YaGNZ0j71vQaLI19q0ubWymbautH00p5KV8owdf4+bf1/NJaPI0zhDUSIjD
Qo01WW31Z9XDSRhKFTnWzmCjBdeIcajyzf10YKsycAW91Itu8aBrMndnQKBgQDb
nNp/JyRwqj0rNljK7DHEs+SD39kHQzzCfqd+dnTPv2sc06+cpym3yu1QcbokULpy
fmRo3bin/pvJQ3aZX/Bdh9woTXqhXDdrrSwWInVYMQPyPk8f/D9mIOJp5FUWMwHD
U+whIZSxsEeE+jtixlWtheKRYkQmzQZXBWdIhYyI3QKBgD+F/6wcZ85QW8nAUyKA
3WrSIx/3cwDgdm4NRGct8Z0ZjTHjiy9ojMOD1L7iMhRQ/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. Buat sertifikat X.509 menggunakan kunci privat 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 berlaku selama 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: xxxxxxxx
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 AWS penyediaan. Itu bisa berupa nama domain internal atau publik.

Anda dapat memeriksa sertifikat dengan perintah berikut:

```
$ cat certificate.pem
```

Output harus berupa string panjang, dengan encode PEM tanpa jeda baris, diawali dengan -----BEGIN CERTIFICATE----- dan diikuti oleh -----END CERTIFICATE-----.

2. Unggah sertifikat X.509 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 menemukannya 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 menemukannya ke bagian komentar untuk organisasi Anda atau bidang “komentar” dari objek di atas.
- Untuk APNIC, kirimkan email sertifikat ke helpdesk@apnic.net untuk menemukannya secara manual ke kolom “pernyataan” untuk rentang alamat Anda. Kirim email menggunakan kontak resmi APNIC untuk alamat IP.

Anda dapat menghapus sertifikat dari catatan RIR Anda setelah tahap penyediaan di bawah ini selesai.

3. 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 ASN 8987 bukan 16509 dan 14618. 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 /56 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 menemukannya 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 AWS Wilayah. 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 memigrasikan iklan dari beban kerja lokal ke tempat 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.

Onboard BYOIP Anda

Proses orientasi untuk BYOIP memiliki 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](#)

Menyediakan rentang alamat yang dapat diiklankan secara publik di AWS

Saat Anda memberikan rentang alamat untuk digunakan AWS, Anda mengonfirmasi bahwa Anda mengontrol rentang alamat dan mengizinkan Amazon untuk mengiklankannya. Kami juga memverifikasi bahwa Anda mengontrol rentang alamat melalui pesan otorisasi yang ditandatangani. Pesan ini ditandatangani dengan key pair X.509 yang ditandatangani sendiri yang Anda gunakan saat memperbarui catatan RDAP dengan sertifikat X.509. AWS memerlukan pesan otorisasi yang ditandatangani secara kriptografis yang disajikan kepada 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 AWS Wilayah di mana rentang BYOIP harus disediakan jika berbeda dari konfigurasi Anda.](#) `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 CIDR IPv6 dari rentang alamat non-publik VPC, CIDR IPv6 hanya dapat diakses melalui opsi konektivitas hybrid yang mendukung IPv6, seperti [AWS Direct Connect](#), [AWS Site-to-Site VPN](#), atau [Amazon VPC Transit Gateway](#).

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
```

Ikhlankan 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 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 mendukung atau memecahkan masalah dengan

andal. Secara khusus, kami tidak dapat menjamin bahwa lalu lintas ke rentang alamat atau sebagian dari rentang akan memasuki jaringan kami.

Untuk meminimalkan waktu henti, Anda dapat mengonfigurasi AWS sumber daya Anda untuk menggunakan alamat dari kumpulan alamat Anda 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
```

Mencabut akses rentang alamat

Untuk berhenti menggunakan rentang alamat Anda AWS, pertama-tama lepaskan alamat IP Elastis apa pun dan lepaskan blok CIDR IPv6 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 AWS, hentikan penyediaan seluruh rentang alamat dan berikan 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.

Menggunakan rentang alamat Anda

Anda dapat melihat serta menggunakan rentang alamat IPv4 dan IPv6 yang telah Anda sediakan di akun Anda.

Rentang alamat IPv4

Anda dapat membuat alamat IP Elastis dari kumpulan alamat IPv4 dan menggunakannya dengan AWS sumber daya Anda, seperti instans EC2, gateway NAT, dan Network Load Balancer.

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.

Validasi BYOIP Anda

1. Validasi pasangan kunci x.509 yang ditandatangani sendiri

Validasi 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
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFd1YiBTZXJ2aWN1czETMBEGA1UEC
wwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpviBXZWIgU2
Vydm1jZXMxEzARBGNVBAsMCKJZT01QIER1bW8xEzARBGNVBAMMCKJZT01QIER1b
W8wggeiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqur9WxkfnanAEskgAseyFypwEEqr4CJijI/5hp9
prh+jswHwWkFRoBRR9FBtwcU/45XDxLga7D3stsI5QeshVRw0aXUdprAnndaTug
mDPkD0vr1475JWDSIm+PUxGwLy+60aBqiaZq35wU/x+wX1AqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdEiW48QaYjhM1UEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rG1HwkJsbhr0VEUyAGu1bwkgcdww
3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBSfFyujN6SYBr2g1HpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2g1HpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBx6nn6YLhz5211fyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0c1r00aFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9wySL507XQz76Uk5cFypB0zbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvphSGCvRD1/qd0/GIDJi77dmZwkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIIsoNPYqrJRzqFU9F3FBjiPJF

```

-----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

```

```
d1YiBTZXJ2aWN1czETMBEGA1UECwwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PS
VAgRGVtbzAeFw0yMTEyMDcyMDI0NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNV
BAYTAk5aMREwDwYDVQQIDAhBdWNrbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDA
aBgNVBAoME0FtYXpvcjBhZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZl
8xEzARBgNVBAMMckJZT01QIERlbW8wggeiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg
gEKAoIBAQCfmacvDp0wZ0ceiXXcR/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanA
EskgAseyFypwEEQr4CJijI/5hp9prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3
stsI5QesHVRw0aXUdprAnndaTugmDPKd0vr1475JWDSIm+PUxGWLy+60aBqiaZq
35wU/x+wX1AqBXg4MZK2KoUu27kYt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp
1ZnVic7NqnhdEiW48QaYjhM1UEfxdaqYUinz8KpjfADZ4Hvqj9jWZ/eXo/9b2r
G1HWkJsbnhr0VEUYAGu1bwkgcdww3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBS
tFyujN6SYBr2g1HpGt0XGF7GbgTAFBgNVHSMEGDAWgBStFyujN6SYBr2g1HpGt0
XGF7GbgTAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwJAA4IBAQBx6nn6Y
Lhz5211fyVfxY0t6o3410bQAEAF08ud+ICtmQ4I04A4B7zV3zIVYr0clr00aFyL
xngwMYN0XY5tVhdQqk4/gmDNEKSZy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9
wySL507XQz76Uk5cFypB0zbnk35UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8
mBGqVpPpey+dXpzzzv1iBKN/VY4ydjgH/LBfdTsVarmmy2vtWBxwrqkFvphSGC
vRD1/qd0/GIDJi77dmZWh/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
LBQAwezELMAkGA1UEBhMCT1oxETAPBgNVBAGMCEf1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFd1YiBTZXJ2aWN1czETMBEGA1UEC
wwKQ11PSVAgRGVtbzETMBEGA1UEAwwKQ11PSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvcjBhZGZlZGZl
ZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZlZGZl
8xEzARBgNVBAMMckJZT01QIERlbW8wggeiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg
gEKAoIBAQCfmacvDp0wZ0ceiXXcR/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanA
EskgAseyFypwEEQr4CJijI/5hp9prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3
stsI5QesHVRw0aXUdprAnndaTugmDPKd0vr1475JWDSIm+PUxGWLy+60aBqiaZq
35wU/x+wX1AqBXg4MZK2KoUu27kYt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp
1ZnVic7NqnhdEiW48QaYjhM1UEfxdaqYUinz8KpjfADZ4Hvqj9jWZ/eXo/9b2r
G1HWkJsbnhr0VEUYAGu1bwkgcdww3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBS
tFyujN6SYBr2g1HpGt0XGF7GbgTAFBgNVHSMEGDAWgBStFyujN6SYBr2g1HpGt0
XGF7GbgTAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwJAA4IBAQBx6nn6Y
Lhz5211fyVfxY0t6o3410bQAEAF08ud+ICtmQ4I04A4B7zV3zIVYr0clr00aFyL
xngwMYN0XY5tVhdQqk4/gmDNEKSZy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9
wySL507XQz76Uk5cFypB0zbnk35UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8
mBGqVpPpey+dXpzzzv1iBKN/VY4ydjgH/LBfdTsVarmmy2vtWBxwrqkFvphSGC
vRD1/qd0/GIDJi77dmZWh/ic90MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoN
PyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

```
R/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QesHVRw0aXUdprAnndaTug
mDPkD0vr1475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wX1AqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdEiW48QaYjhM1UEf
xdaqYUinz8KpjfADZ4Hvqj9jWZ/eXo/9b2rG1HWkJsbhr0VEUyAGu1bwkgcdww
3A7Nj0xQbAgMBAAGjUzBRMB0GA1UdDgQWBBSstFyujN6SYBr2g1HpGt0XGF7GbGT
AfBgNVHSMEGDAWgBSstFyujN6SYBr2g1HpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIB3DQEBCwJAA4IBAQBx6nn6YLhZ5211fyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0c1r00aFyLxngwMYN0XY5tVhdQqk4/gmDNEKS
Zy2QkX4Eg0YUWVz0yt6fPzj0vJLcsqc1hcF9wySL507XQz76Uk5cFypB0zbnk35
UkWrzaA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvphSGCvRD1/qd0/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIs0NPYqrJRzqFU9F3FBjiPJF
-----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.

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.

Ketersediaan wilayah

Fitur BYOIP saat ini tersedia di semua [Wilayah AWS](#) komersial kecuali untuk Wilayah Tiongkok.

Ketersediaan Local Zone

[Zona Lokal](#) adalah perpanjangan dari AWS Wilayah dalam kedekatan geografis dengan pengguna Anda. Local Zones dikelompokkan ke dalam “grup perbatasan jaringan”. Di AWS, grup perbatasan jaringan adalah kumpulan Availability Zones (AZ), Local Zones, atau Wavelength Zones tempat

mengiklankan alamat IP publik. AWS Local Zones mungkin memiliki grup perbatasan jaringan yang berbeda dari AZ di suatu AWS Wilayah 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`:

- us-east-1-dfw-2
- us-west-2-lax-1
- us-west-2-phx-2

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 saat ini tidak dapat menyediakan atau mengiklankan rentang alamat BYOIPv6 di Local Zones.

Pelajari selengkapnya

Untuk informasi lebih lanjut, lihat AWS Online Tech talk [Deep Dive on Bring Your Own IP](#).

Alamat IP elastis

Alamat IP Elastis adalah sebuah alamat IPv4 statis untuk komputasi cloud dinamis. Alamat IP Elastis 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 instans Anda tidak memiliki alamat IPv4 publik, Anda dapat mengaitkan alamat IP Elastis dengan instans Anda

untuk mengaktifkan komunikasi dengan internet. Misalnya, ini memungkinkan Anda untuk terhubung ke instans Anda dari komputer lokal Anda.

Daftar Isi

- [Harga alamat IP Elastis](#)
- [Dasar alamat IP Elastis](#)
- [Cara menggunakan alamat IP Elastis](#)
- [Kuota alamat IP Elastis](#)

Harga alamat IP Elastis

AWS mengenakan biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastis. 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 Elastis berasal dari kumpulan alamat IPv4 Amazon, atau dari kumpulan alamat IPv4 khusus yang telah Anda bawa ke akun Anda. AWS
- 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 Elastis dengan instans 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 Elastis dikaitkan dengan instans sebagai gantinya. Anda tidak dapat menggunakan kembali alamat IPv4 publik yang sebelumnya terkait dengan instans dan Anda tidak dapat mengonversi alamat IPv4 publik tersebut menjadi alamat IP Elastis. 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 melepaskannya. Anda dikenakan biaya untuk semua alamat IP Elastic di akun Anda, terlepas dari apakah alamat tersebut terkait atau tidak terkait dengan instans. 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 Elastis dari kumpulan alamat IP yang telah Anda bawa ke AWS akun Anda, itu tidak dihitung terhadap batas alamat IP Elastis 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.

Cara menggunakan alamat IP Elastis

Bagian berikut ini menjelaskan bagaimana Anda bekerja dengan alamat IP Elastis.

Tugas

- [Mengalokasikan alamat IP Elastis](#)
- [Menjelaskan alamat IP Elastis Anda](#)
- [Menandai alamat IP Elastis](#)
- [Kaitkan alamat IP Elastis dengan instans atau antarmuka jaringan](#)
- [Pisahkan alamat IP Elastis](#)

- [Transfer alamat IP Elastis](#)
- [Melepas alamat Elastic IP](#)
- [Memulihkan alamat IP Elastis](#)
- [Menggunakan DNS terbalik untuk aplikasi email](#)

Mengalokasikan alamat IP Elastis

Anda dapat mengalokasikan alamat IP Elastis dari kumpulan alamat IPv4 publik Amazon, atau dari kumpulan alamat IP khusus yang telah Anda bawa ke akun Anda. AWS Untuk informasi selengkapnya tentang membawa rentang alamat IP Anda sendiri ke AWS akun Anda, lihat [Bring your own IP addresses \(BYOIP\) di Amazon EC2](#).

Anda dapat mengalokasikan alamat IP Elastis menggunakan salah satu metode berikut.

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 Availability Zones (AZ), Local Zones, atau Wavelength Zones yang mengiklankan alamat IP publik. AWS Local Zones dan Wavelength Zones mungkin memiliki grup perbatasan jaringan yang berbeda dari AZ di Wilayah untuk memastikan latensi minimum atau jarak fisik antara jaringan dan pelanggan yang mengakses sumber daya AWS 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 Availability Zone, Local Zones, atau Wavelength Zone 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 dari kumpulan alamat IP yang telah Anda bawa ke akun Anda. AWS Opsi ini dinonaktifkan jika Anda tidak memiliki kumpulan alamat IP.
- Kumpulan alamat IPv4 milik pelanggan—Jika Anda ingin mengalokasikan alamat IPv4 dari kumpulan alamat yang dibuat dari jaringan on-premise Anda untuk digunakan dengan AWS Outpost. Opsi ini dinonaktifkan jika Anda tidak memiliki AWS Outpost.

6. (Opsional) Tambahkan atau hapus tanda.

[Menambahkan tanda] Pilih Tambahkan tanda baru dan lakukan hal berikut:

- Untuk Kunci, masukkan nama kunci.
- Untuk Nilai, masukkan nilai kunci.

[Hapus tanda] Pilih Hapus di sebelah kanan Kunci dan Nilai tanda.

7. Pilih Alokasikan.

AWS CLI

Untuk mengalokasikan Alamat IP elastis

Gunakan perintah AWS CLI [allocate-address](#).

PowerShell

Untuk mengalokasikan Alamat IP elastis

Gunakan perintah [New-EC2Address](#) AWS Tools for Windows PowerShell .

Menjelaskan alamat IP Elastis Anda

Anda dapat menjelaskan alamat IP Elastis menggunakan salah satu metode berikut.

Console

Untuk menjelaskan alamat IP Elastis Anda

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 melihat dan memilih Tindakan, Lihat detail.

AWS CLI

Untuk menjelaskan alamat IP Elastis Anda

Gunakan [perintah deskripsi-alamat](#) AWS CLI .

PowerShell

Untuk menjelaskan alamat IP Elastis Anda

Gunakan perintah [Get-EC2Address](#) AWS Tools for Windows PowerShell .

Menandai alamat IP Elastis

Anda dapat menetapkan tanda kustom ke alamat IP Elastis Anda untuk mengategorikannya dengan cara berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Ini membantu Anda menemukan dengan cepat alamat IP Elastis spesifik berdasarkan tanda kustom yang Anda tetapkan.

Pelacakan alokasi biaya menggunakan tanda alamat IP Elastis tidak didukung.

Anda dapat menandai alamat IP Elastis menggunakan salah satu metode berikut.

Console

Untuk menandai 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 untuk menandai dan memilih Tindakan, Lihat detail.
4. Di bagian Tanda, pilih Kelola tanda.
5. Tentukan tombol tanda dan pasangan nilai.
6. (Opsional) Pilih Tambahkan tanda untuk menambahkan tanda tambahan.
7. Pilih Simpan.

AWS CLI

Untuk memberi tag alamat Elastic IP

Gunakan perintah [create-tags](#) AWS CLI .

```
aws ec2 create-tags --resources eipalloc-12345678 --tags Key=Owner,Value=TeamA
```

PowerShell

Untuk menandai alamat IP Elastis

Gunakan perintah [New-EC2Tag](#) AWS Tools for Windows PowerShell .

Perintah New-EC2Tag memerlukan parameter Tag, yang menentukan pasangan kunci dan nilai untuk digunakan untuk tanda alamat IP Elastis. Perintah berikut akan membuat parameter Tag.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag  
PS C:\> $tag.Key = "Owner"  
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource eipalloc-12345678 -Tag $tag
```

Kaitkan alamat IP Elastis dengan instans atau antarmuka jaringan

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 lebih lanjut, lihat [Gateway internet](#) di Panduan Pengguna Amazon VPC.

Anda dapat mengaitkan alamat IP Elastis dengan instans atau antarmuka jaringan menggunakan salah satu metode berikut.

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 [asosiasi-alamat](#) AWS CLI .

PowerShell

Untuk mengaitkan alamat IP Elastis

Gunakan perintah [Register-EC2Address](#) AWS Tools for Windows PowerShell .

Pisahkan alamat IP Elastis

: Untuk melepaskan pengaitan alamat IP Elastis dari instans atau antarmuka jaringan. Setelah Anda memisahkan alamat IP Elastis, Anda dapat mengaitkan kembali dengan sumber daya lain.

Anda dapat memisahkan alamat IP Elastis menggunakan salah satu metode berikut.

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 .

PowerShell

Untuk memisahkan alamat IP Elastis

Gunakan [Unregister-EC2Address](#) AWS Tools for Windows PowerShell perintah.

Transfer alamat IP Elastis

Bagian ini menjelaskan cara mentransfer alamat IP Elastis dari satu Akun AWS ke yang lain.

Mentransfer alamat IP Elastis dapat membantu dalam situasi berikut:

- **Restrukturisasi organisasi** — Gunakan transfer alamat IP Elastis untuk memindahkan beban kerja dengan cepat dari satu Akun AWS ke yang lain. Anda tidak perlu menunggu alamat IP Elastis baru diizinkan terdaftar di grup keamanan dan NACL Anda.
- **Administrasi keamanan terpusat** — Gunakan akun AWS keamanan terpusat untuk melacak dan mentransfer alamat IP Elastis yang telah diperiksa untuk kepatuhan keamanan.
- **Pemulihan bencana** - Gunakan transfer alamat IP Elastis untuk memetakan ulang IP dengan cepat untuk beban kerja internet yang dihadapi publik selama peristiwa darurat.

Tidak ada biaya untuk mentransfer alamat IP Elastis.

Tugas

- [Aktifkan transfer alamat IP Elastis](#)
- [Nonaktifkan transfer alamat IP Elastis](#)
- [Menerima alamat IP Elastis yang ditransfer](#)

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 Elastis dari Akun AWS (akun sumber) apa pun ke AWS akun lain di AWS Wilayah yang sama (akun transfer).
- 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 itu, akun sumber dapat melihat transfer yang tertunda (misalnya di AWS konsol atau dengan menggunakan [perintah AWS CLI deskripsi-alamat-transfer](#)). Setelah tujuh hari, transfer berakhir dan kepemilikan alamat IP Elastis kembali ke akun sumber.
- Transfer yang diterima dapat dilihat oleh akun sumber (misalnya di AWS konsol atau dengan menggunakan [AWS CLI perintah deskripsi-alamat-transfer](#)) selama tiga hari setelah transfer diterima.

- AWS tidak memberi tahu akun transfer tentang permintaan transfer alamat IP Elastis 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 Elastis yang dialokasikan dari kumpulan alamat IPv4 publik yang Anda bawa ke kolam alamat Bring Your Own IP (BYOIP). Akun AWS
- 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 Elastis 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 IP milik pelanggan](#) di Panduan Pengguna AWS Outposts .
- 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 Elastis ditransfer ke Akun AWS di luar organisasi, riwayat audit IPAM dari alamat IP Elastis akan 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 Elastis ke satu AWS akun.

- Pilih Beberapa akun jika Anda mentransfer alamat IP Elastis 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](#).

PowerShell

Untuk mengaktifkan transfer alamat IP Elastis

Gunakan perintah [Enable-EC2AddressTransfer](#).

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 AWS akun sumber.
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](#).

PowerShell

Untuk menonaktifkan transfer alamat IP Elastis

Gunakan perintah [Disable-EC2AddressTransfer](#).

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 itu, akun sumber dapat melihat transfer yang tertunda (misalnya di AWS konsol atau dengan menggunakan [perintah AWS CLI deskripsi-alamat-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:

- **AddressLimitTerlampau:** Jika akun transfer Anda telah melebihi kuota alamat IP Elastis, akun sumber dapat mengaktifkan transfer alamat IP Elastis, tetapi pengecualian ini terjadi ketika akun transfer mencoba menerima transfer. Secara default, semua Akun AWS 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 Elastis yang Anda coba transfer untuk menggunakan pencarian DNS terbalik, akun sumber dapat mengaktifkan transfer untuk alamat IP Elastis, 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 [Menggunakan DNS terbalik untuk aplikasi email](#).
- **InvalidTransfer. AddressAssociated:** Jika alamat IP Elastis dikaitkan dengan instans ENI atau EC2, akun sumber dapat mengaktifkan transfer untuk alamat IP Elastis, 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 AWS Support](#).

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](#).

PowerShell

Untuk menerima transfer alamat IP Elastis

Gunakan perintah [Approve-EC2AddressTransfer](#).

Melepas alamat Elastic IP

Jika Anda tidak lagi memerlukan alamat IP Elastis, kami sarankan Anda menggunakan salah satu metode berikut. Alamat yang akan dirilis saat ini tidak boleh dikaitkan dengan AWS sumber daya, seperti instans EC2, gateway NAT, atau Network Load Balancer.

Note

Jika Anda menghubungi AWS dukungan untuk mengatur DNS terbalik untuk alamat Elastic IP (EIP), Anda dapat menghapus DNS terbalik, tetapi Anda tidak dapat melepaskan alamat IP Elastis karena telah dikunci oleh dukungan. AWS Untuk membuka kunci alamat IP Elastis, hubungi [AWS Support](#). Setelah alamat IP elastis dibuka, Anda dapat melepaskan alamat IP Elastis.

Console

Untuk melepas alamat Elastic IP

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 .

PowerShell

Untuk merilis alamat IP Elastis

Gunakan perintah [Remove-EC2Address](#) AWS Tools for Windows PowerShell .

Memulihkan alamat IP Elastis

Jika Anda telah melepas alamat IP Elastis Anda, Anda dapat memulihkannya. Aturan-aturan berikut berlaku:

- Anda tidak dapat memulihkan alamat IP Elastis jika sudah dialokasikan ke akun AWS lain, atau jika mengakibatkan melebihi batas alamat IP Elastis Anda.
- Anda tidak dapat memulihkan tanda terkait dengan alamat IP Elastis.
- Anda dapat memulihkan alamat IP Elastis menggunakan Amazon EC2 API atau alat baris perintah saja.

AWS CLI

Untuk memulihkan alamat Elastic IP

Gunakan AWS CLI perintah [allocate-address](#) dan tentukan alamat IP menggunakan parameter sebagai berikut. `--address`

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```

PowerShell

Untuk memulihkan alamat IP Elastis

Gunakan [New-EC2Address](#) AWS Tools for Windows PowerShell perintah dan tentukan alamat IP menggunakan `-Address` parameter sebagai berikut.

```
PS C:\> New-EC2Address -Address 203.0.113.3 -Domain vpc -Region us-east-1
```

Menggunakan DNS terbalik untuk aplikasi email

Jika Anda bermaksud mengirimkan email ke pihak ketiga dari sebuah instans, kami sarankan agar Anda memberikan satu atau beberapa alamat IP Elastis dan menetapkan data DNS terbalik statis ke alamat IP Elastis 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

- Sebelum Anda membuat catatan DNS terbalik, Anda harus mengatur catatan DNS lanjutan yang sesuai (tipe catatan A) yang menunjuk ke alamat IP Elastis Anda.

- 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.
- AWS GovCloud (US) Region

Anda tidak dapat membuat rekaman DNS terbalik menggunakan konsol atau AWS CLI. AWS harus menetapkan catatan DNS terbalik statis untuk Anda. Buka [Permintaan untuk menghapus batasan pengiriman DNS dan email terbalik](#) dan berikan kami alamat IP Elastis Anda dan catatan DNS terbalik.

Membuat catatan DNS terbalik

Untuk membuat catatan DNS terbalik, pilih tab yang cocok dengan metode pilihan Anda.

Console

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

Gunakan [modify-address-attribute](#) perintah di AWS CLI, seperti yang ditunjukkan pada contoh berikut:

```
aws ec2 modify-address-attribute --allocation-id eipalloc-abcdef01234567890 --  
domain-name example.com  
{  
  "Addresses": [  
    {  
      "PublicIp": "192.0.2.0",  
      "AllocationId": "eipalloc-abcdef01234567890",  
      "PtrRecord": "example.net."  
      "PtrRecordUpdate": {  
        "Value": "example.com.",  
        "Status": "PENDING"  
      }  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Membuat catatan DNS terbalik

Untuk menghapus catatan DNS terbalik, pilih tab yang cocok dengan metode pilihan Anda.

Console

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

Gunakan [reset-address-attribute](#) perintah di AWS CLI, seperti yang ditunjukkan pada contoh berikut:

```
aws ec2 reset-address-attribute --allocation-id eipalloc-abcdef01234567890 --  
attribute domain-name  
{  
  "Addresses": [  
    {  
      "PublicIp": "192.0.2.0",  
      "AllocationId": "eipalloc-abcdef01234567890",  
      "PtrRecord": "example.com."  
      "PtrRecordUpdate": {  
        "Value": "example.net.",  
        "Status": "PENDING"  
      }  
    }  
  ]  
}
```

Note

Jika Anda menerima kesalahan berikut saat menjalankan perintah, Anda dapat mengirimkan [Permintaan untuk menghapus batasan pengiriman email](#) AWS Support untuk bantuan.

Alamat dengan id alokasi tidak dapat dirilis karena dikunci ke akun Anda.

Kuota alamat IP Elastis

Secara default, semua AWS akun memiliki kuota lima (5) alamat IP elastis per Wilayah, karena alamat internet publik (IPv4) adalah sumber daya publik yang langka. Kami sangat menganjurkan Anda untuk menggunakan alamat IP Elastis terutama untuk kemampuan memetakan ulang alamat ke instans lain jika terjadi kegagalan instans, dan untuk menggunakan [nama host DNS](#) untuk semua komunikasi antar simpul lainnya.

Untuk memverifikasi berapa banyak alamat IP Elastis yang digunakan

Buka konsol Amazon EC2 di [AWS.amazon.com/EC2/](https://aws.amazon.com/EC2/) dan pilih Elastic IP dari panel navigasi.

Guna memverifikasi kuota akun saat ini untuk alamat IP Elastis

1. Buka konsol Kuota Layanan di <https://console.aws.amazon.com/servicequotas/>.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah.
3. Pada Dasbor, pilih Amazon Elastic Compute Cloud (Amazon EC2).

Jika Amazon Elastic Compute Cloud (Amazon EC2) tidak tercantum di Dasbor, pilih Layanan AWS, masukkan **EC2** di bidang pencarian, lalu pilih Amazon Elastic Compute Cloud (Amazon EC2).

4. Di halaman kuota layanan Amazon EC2, masukkan **IP** di bidang pencarian. Batasnya adalah IP Elastis EC2-VPC. Untuk informasi lebih lanjut, pilih batasnya.

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](#).

Antarmuka jaringan elastis

Antarmuka jaringan elastis adalah komponen jaringan logis dalam VPC yang mewakili kartu jaringan virtual. Ini dapat mencakup atribut berikut:

- Alamat IPv4 privat primer dari rentang alamat IPv4 VPC Anda
- Alamat IPv6 primer dari rentang alamat IPv6 VPC Anda
- Satu atau lebih alamat IPv4 privat sekunder dari rentang alamat IPv4 VPC Anda
- Satu alamat IP Elastis (IPv4) per alamat IPv4 privat
- Satu alamat IPv4 publik
- Satu atau beberapa alamat IPv6
- Satu atau lebih grup keamanan
- Alamat MAC
- Penanda cek sumber/tujuan
- Deskripsi

Anda dapat membuat dan mengonfigurasi antarmuka jaringan dan melampirkannya ke instans di Zona Ketersediaan yang sama. Akun Anda mungkin juga memiliki antarmuka jaringan yang dikelola pemohon, yang dibuat dan dikelola oleh AWS layanan untuk memungkinkan Anda menggunakan sumber daya dan layanan lain. Anda tidak dapat mengelola antarmuka jaringan ini sendiri. Untuk informasi selengkapnya, lihat [Antarmuka jaringan yang dikelola pemohon](#).

AWS Sumber daya ini disebut sebagai antarmuka jaringan di AWS Management Console 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".

Isi

- [Dasar-dasar antarmuka jaringan](#)
- [Kartu jaringan](#)
- [Alamat IP per antarmuka jaringan per tipe instans](#)
- [Bekerja dengan antarmuka jaringan](#)
- [Praktik terbaik untuk mengonfigurasi antarmuka jaringan](#)

- [Skenario untuk antarmuka jaringan](#)
- [Antarmuka jaringan yang dikelola pemohon](#)
- [Menetapkan prefiks ke antarmuka jaringan Amazon EC2](#)

Dasar-dasar antarmuka jaringan

Anda dapat membuat antarmuka jaringan, melampirkannya ke sebuah instans, melepaskannya dari sebuah instans, dan melampirkannya ke instans lain. Atribut antarmuka jaringan mengikutinya saat dilampirkan atau dilepaskan dari sebuah instans dan dilampirkan kembali ke instans lain. Saat Anda memindahkan antarmuka jaringan dari satu instans ke instans lainnya, lalu lintas jaringan dialihkan ke instans baru.

Antarmuka jaringan primer

Setiap instans memiliki antarmuka jaringan default, yang disebut antarmuka jaringan primer. Anda tidak dapat melepaskan antarmuka jaringan primer dari sebuah instans. Anda dapat membuat dan memasang antarmuka jaringan tambahan. Jumlah maksimum antarmuka jaringan yang dapat Anda gunakan bervariasi menurut tipe instans. Untuk informasi selengkapnya, lihat [Alamat IP per antarmuka jaringan per tipe instans](#).

Alamat IPv4 publik untuk antarmuka jaringan

Di VPC, 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) ditetapkan sebagai alamat IPv4 publik. Untuk informasi selengkapnya, lihat [Pengaturan subnet](#) di Panduan Pengguna Amazon VPC. Alamat IPv4 publik ditetapkan dari kumpulan alamat IPv4 publik Amazon. Saat Anda meluncurkan sebuah instans, alamat IP ditetapkan ke antarmuka jaringan primer yang dibuat.

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. Jika Anda meluncurkan sebuah instans dan menentukan antarmuka jaringan yang ada sebagai antarmuka jaringan primer, atribut alamat IPv4 publik ditentukan oleh antarmuka jaringan ini.

Untuk informasi selengkapnya, lihat [Alamat IPv4 publik](#).

Alamat IP elastis untuk antarmuka jaringan

Jika Anda memiliki alamat IP Elastis, Anda dapat mengaitkannya dengan salah satu alamat IPv4 privat untuk antarmuka jaringan. Anda dapat mengaitkan satu alamat IP Elastis dengan setiap alamat IPv4 privat.

Jika Anda memisahkan alamat IP Elastis dari antarmuka jaringan, Anda dapat melepaskannya kembali ke kumpulan alamat. Ini adalah satu-satunya cara untuk mengaitkan alamat IP Elastis dengan sebuah instans di subnet atau VPC yang berbeda, karena antarmuka jaringan dikhususkan untuk subnet.

Alamat IPv6 untuk antarmuka jaringan

Jika Anda mengasosiasikan blok CIDR IPv6 dengan VPC dan subnet Anda, Anda dapat menetapkan satu atau lebih 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. Untuk informasi selengkapnya, lihat [Pengaturan subnet](#) di Panduan Pengguna Amazon VPC. Saat Anda meluncurkan sebuah instans, alamat IPv6 ditetapkan ke antarmuka jaringan primer yang dibuat.

Untuk informasi selengkapnya, lihat [Alamat IPv6](#).

Delegasi Prefiks

Prefiks Delegasi prefiks adalah rentang CIDR IPv4 atau IPv6 privat yang dicadangkan yang Anda alokasikan untuk penugasan otomatis atau manual ke antarmuka jaringan yang terkait dengan instans. Dengan menggunakan Prefiks yang Didelegasikan, Anda dapat meluncurkan layanan lebih cepat dengan menetapkan berbagai alamat IP sebagai prefiks tunggal.

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.

Pemeriksaan sumber / tujuan

Anda dapat mengaktifkan atau menonaktifkan pemeriksaan sumber/tujuan, yang memastikan bahwa instans adalah salah satu sumber atau tujuan dari setiap lalu lintas yang diterimanya. Pemeriksaan sumber/tujuan diaktifkan secara default. Anda harus menonaktifkan pemeriksaan sumber/tujuan jika instans menjalankan layanan seperti terjemahan alamat jaringan, perutean, atau firewall.

Memantau lalu lintas IP

Anda dapat mengaktifkan log aliran VPC di antarmuka jaringan Anda untuk menangkap informasi tentang lalu lintas IP yang pergi ke dan dari antarmuka jaringan. Setelah membuat log aliran, Anda dapat melihat dan mengambil datanya di Amazon CloudWatch Logs. Untuk informasi selengkapnya, lihat [Log Alur VPC](#) di Panduan Pengguna Amazon VPC.

Penetapan alamat IPv4 publik otomatis

Anda dapat mengaktifkan dan menonaktifkan penetapan otomatis alamat IPv4 publik ke antarmuka jaringan. Opsi ini dapat diaktifkan untuk antarmuka jaringan apa pun tetapi hanya akan berlaku untuk antarmuka jaringan utama (eth0). Untuk informasi selengkapnya, lihat [Mengelola alamat IP](#).

Kartu jaringan

Instans dengan beberapa kartu jaringan memberikan performa jaringan yang lebih tinggi, termasuk kemampuan bandwidth di atas 100 Gbps dan performa kecepatan paket yang ditingkatkan. Setiap antarmuka jaringan terpasang ke kartu jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0.

Jika Anda mengaktifkan Elastic Fabric Adapter (EFA) saat Anda meluncurkan instans yang mendukung beberapa kartu jaringan, semua kartu jaringan tersedia. Anda dapat menetapkan hingga satu EFA per kartu jaringan. EFA dihitung sebagai antarmuka jaringan.

Instans berikut mendukung beberapa kartu jaringan. Semua tipe instans lainnya mendukung satu kartu jaringan.

Jenis instans	Jumlah kartu jaringan
c6in.32xlarge	2
c6in.metal	2
d11.24xlarge	4
hpc6id.32xlarge	2
hpc7a.12xlarge	2
hpc7a.24xlarge	2

Jenis instans	Jumlah kartu jaringan
hpc7a.48xlarge	2
hpc7a.96xlarge	2
m6idn.32xlarge	2
m6idn.metal	2
m6in.32xlarge	2
m6in.metal	2
p4d.24xlarge	4
p4de.24xlarge	4
p5.48xlarge	32
r6idn.32xlarge	2
r6idn.metal	2
r6in.32xlarge	2
r6in.metal	2
trn1.32xlarge	8
trn1n.32xlarge	16
u7in-16tb.224xlarge	2
u7in-24tb.224xlarge	2
u7in-32tb.224xlarge	2

Alamat IP per antarmuka jaringan per tipe instans

Setiap jenis instans 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. Tidak semua tipe instans mendukung pengalamatan IPv6.

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 - Komputasi kinerja tinggi](#)
- [Spesifikasi jaringan — Generasi sebelumnya](#)

Untuk mengambil informasi antarmuka jaringan menggunakan AWS CLI

Anda dapat menggunakan AWS CLI 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 C5.

```
aws ec2 describe-instance-types --filters "Name=instance-type,Values=c5.*" --query
"InstanceTypes[].{Type: InstanceType, MaxENI: NetworkInfo.MaximumNetworkInterfaces,
IPv4addr: NetworkInfo.Ipv4AddressesPerInterface}" --output table
```

```
-----
|           DescribeInstanceTypes           |
+-----+-----+-----+
| IPv4addr | MaxENI | Type |
+-----+-----+-----+
| 30       | 8      | c5.4xlarge |
| 50       | 15     | c5.24xlarge |
| 15       | 4      | c5.xlarge |
| 30       | 8      | c5.12xlarge |
-----
```

10	3	c5.large	
15	4	c5.2xlarge	
50	15	c5.metal	
30	8	c5.9xlarge	
50	15	c5.18xlarge	
+-----+	+-----+	+-----+	+-----+

Bekerja dengan antarmuka jaringan

Anda dapat bekerja dengan antarmuka jaringan menggunakan konsol Amazon EC2 atau baris perintah.

Daftar Isi

- [Membuat antarmuka jaringan](#)
- [Melihat detail tentang antarmuka jaringan](#)
- [Melampirkan antarmuka jaringan ke sebuah instans](#)
- [Melepaskan antarmuka jaringan dari sebuah instans](#)
- [Mengelola alamat IP](#)
- [Mengubah atribut antarmuka jaringan](#)
- [Menambahkan atau mengedit tanda](#)
- [Menghapus antarmuka jaringan](#)

Membuat antarmuka jaringan

Anda dapat membuat antarmuka jaringan di subnet. Anda tidak dapat memindahkan antarmuka jaringan ke subnet lain setelah dibuat. Anda harus memasang antarmuka jaringan ke sebuah instans di Zona Ketersediaan yang sama.

Untuk membuat antarmuka jaringan menggunakan konsol

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 di langkah selanjutnya berubah tergantung pada tipe subnet yang Anda pilih (hanya IPv4, hanya IPv6, atau dual-stack (IPv4 dan IPv6)).

6. Untuk Alamat IPv4 privat, lakukan salah satu langkah berikut:
 - Pilih Menetapkan secara otomatis untuk mengizinkan Amazon EC2 untuk memilih alamat IPv4 dari subnet.
 - Pilih Kustom dan masukkan alamat IPv4 yang Anda pilih dari subnet.
7. (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 Menetapkan secara otomatis untuk mengizinkan Amazon EC2 untuk memilih alamat IPv6 dari subnet.
 - Pilih Kustom dan masukkan alamat IPv6 yang Anda pilih dari subnet.
8. (Opsional) Jika Anda membuat antarmuka jaringan dalam subnet dual-stack atau IPv6 saja, Anda memiliki opsi untuk Menetapkan IP IPv6 Primer. 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 ENI ini akan dilampirkan bergantung pada alamat IPv6-nya yang tidak berubah. AWS akan secara otomatis menetapkan alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instans Anda untuk menjadi alamat IPv6 utama. Setelah Anda mengaktifkan alamat IPv6 GUA 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.
9. (Opsional) Untuk membuat Elastic Fabric Adapter, pilih Elastic Fabric Adapter, Aktifkan.
10. (Opsional) Di bawah Pengaturan lanjutan, untuk Batas waktu pelacakan koneksi idle, modifikasi batas waktu koneksi idle default. Untuk informasi selengkapnya tentang opsi ini, 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 (5 hari). Default: 432.000 detik. 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 (3 menit). Default: 180 detik.

11. Untuk Grup keamanan, pilih satu atau beberapa grup keamanan.
12. (Opsional) Untuk setiap tanda, pilih Tambahkan tanda baru dan masukkan kunci tanda dan nilai tanda opsional tersebut.
13. Pilih Buat antarmuka jaringan.

Untuk membuat antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [create-network-interface](#) (AWS CLI)
- [New-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Melihat detail tentang antarmuka jaringan

Anda dapat melihat semua antarmuka jaringan di akun Anda.

Untuk mendeskripsikan antarmuka jaringan menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Untuk melihat halaman rincian antarmuka jaringan, pilih ID antarmuka jaringan. Sebagai alternatif, untuk melihat informasi tanpa meninggalkan halaman antarmuka jaringan, pilih kotak centang untuk antarmuka jaringan.

Untuk menggambarkan antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-network-interfaces](#) (AWS CLI)
- [Get-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Untuk mendeskripsikan atribut antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-network-interface-attribute](#) (AWS CLI)
- [Get-EC2NetworkInterfaceAttribute](#) (AWS Tools for Windows PowerShell)

Melampirkan antarmuka jaringan ke sebuah instans

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](#).

Important

Untuk instans EC2 di subnet hanya IPv6, jika Anda melampirkan antarmuka jaringan sekunder ke instans, nama host DNS privat dari antarmuka jaringan kedua akan diselesaikan ke alamat IPv6 pertama pada antarmuka jaringan pertama instans. Untuk informasi selengkapnya tentang nama host DNS privat instans EC2, lihat [Tipe nama host instans Amazon EC2](#).

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](#).

Instances page

Untuk memasang antarmuka jaringan ke sebuah instans 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. Jika Anda melampirkan antarmuka jaringan sekunder ke instance, antarmuka jaringan dapat berada di VPC yang sama dengan instance Anda atau di VPC lain yang Anda miliki (selama antarmuka jaringan berada di subnet yang berada di Availability Zone yang sama dengan instance Anda). 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.

Network Interfaces page

Untuk memasang antarmuka jaringan ke sebuah instans 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.

Untuk memasang antarmuka jaringan ke sebuah instans menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

Note

Anda dapat melampirkan antarmuka jaringan yang ada di VPC lain (tetapi di Availability Zone yang sama) ke instance menggunakan perintah [AWS CLI `attach-network-interface`](#). Anda tidak dapat melakukan ini menggunakan AWS Management Console.

- [attach-network-interface](#) (AWS CLI)
- [Add-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Melepaskan antarmuka jaringan dari sebuah instans

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, WorkSpace a, 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.

Instances page

Untuk melepaskan antarmuka jaringan dari sebuah instans 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.

Network Interfaces page

Untuk melepaskan antarmuka jaringan dari sebuah instans 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.

Untuk melepaskan antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [detach-network-interface](#) (AWS CLI)
- [Dismount-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Mengelola alamat IP

Anda dapat mengelola alamat IP berikut untuk antarmuka jaringan Anda:

- Alamat IP Elastis (satu per alamat IPv4 privat)
- Alamat IPv4
- Alamat IPv6
- Alamat IPv6 primer

Untuk mengelola alamat IP Elastis dari antarmuka jaringan menggunakan konsol

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 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.
5. 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.

Untuk mengelola alamat IPv4 dan IPv6 antarmuka jaringan menggunakan konsol

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 Tindakan, Kelola Alamat IP.
5. Bentangkan antarmuka jaringan.
6. Untuk Alamat IPv4, ubah alamat IP sesuai kebutuhan. Untuk menetapkan alamat IPv4, pilih Tetapkan alamat IP baru dan kemudian tentukan alamat IPv4 dari rentang subnet atau biarkan pilih satu untuk Anda. AWS Untuk membatalkan penetapan alamat IPv4, pilih Batalkan penetapan di samping alamat.
7. Untuk menetapkan atau membatalkan penetapan alamat IPv4 publik ke antarmuka jaringan, pilih Auto-assign IP publik. Opsi ini dapat diaktifkan atau dinonaktifkan untuk antarmuka jaringan apa pun tetapi hanya akan berlaku untuk antarmuka jaringan utama (eth0).
8. Untuk Alamat IPv6, ubah alamat IP sesuai kebutuhan. Untuk menetapkan alamat IPv6, pilih Tetapkan alamat IP baru dan kemudian tentukan alamat IPv6 dari rentang subnet atau biarkan pilih satu untuk Anda. AWS Untuk membatalkan penetapan alamat IPv6, pilih Batalkan penetapan di samping alamat.
9. (Opsional) Jika Anda memodifikasi antarmuka jaringan dalam subnet dual-stack atau IPv6 saja, Anda memiliki opsi untuk Menetapkan IP IPv6 Primer. Menetapkan alamat IPv6 primer memungkinkan Anda untuk menghindari mengganggu lalu lintas ke instans atau ENI. Pilih Aktifkan jika instance ENI ini akan dilampirkan bergantung pada alamat IPv6-nya yang tidak berubah. AWS akan secara otomatis menetapkan alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instans Anda untuk menjadi alamat IPv6 utama. Setelah Anda mengaktifkan alamat IPv6 GUA 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. Pilih Simpan.

Untuk mengelola alamat IP antarmuka jaringan menggunakan AWS CLI

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi lebih lanjut tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [assign-ipv6-addresses](#)
- [associate-address](#)
- [disassociate-address](#)
- [unassign-ipv6-addresses](#)

Untuk mengelola alamat IP dari antarmuka jaringan menggunakan Alat untuk Windows PowerShell

Anda dapat menggunakan salah satu perintah berikut ini.

- [Register-EC2Address](#)
- [Register-EC2Ipv6 AddressList](#)
- [Unregister-EC2Address](#)
- [Unregister-EC2Ipv6 AddressList](#)

Mengubah atribut antarmuka jaringan

Anda dapat mengubah atribut antarmuka jaringan berikut:

- [Deskripsi](#)
- [Grup keamanan](#)
- [Hapus saat penghentian](#)
- [Pemeriksaan sumber/tujuan](#)

Untuk mengubah deskripsi antarmuka jaringan menggunakan konsol

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, Ubah deskripsi.

5. Untuk Deskripsi, masukkan deskripsi untuk antarmuka jaringan.
6. Pilih Simpan.

Untuk mengubah grup keamanan dari antarmuka jaringan menggunakan konsol

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, Ubah grup keamanan.
5. Untuk Grup keamanan terkait, pilih grup keamanan yang akan digunakan, lalu pilih Simpan.

Grup keamanan dan antarmuka jaringan harus dibuat untuk VPC yang sama. Untuk mengubah grup keamanan untuk antarmuka yang dimiliki oleh layanan lain, seperti Elastic Load Balancing, lakukan melalui layanan tersebut.

Untuk mengubah perilaku penghentian antarmuka jaringan menggunakan konsol

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, Ubah perilaku penghentian.
5. Pilih atau hapus Hapus saat penghentian, Aktifkan sesuai kebutuhan, lalu pilih Simpan.

Untuk mengubah sumber/tujuan memeriksa antarmuka jaringan menggunakan konsol

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, Ubah pemeriksaan source/dest.
5. Pilih atau hapus Pemeriksaan sumber / tujuan, Aktifkan sesuai kebutuhan, lalu pilih Simpan.

Untuk mengubah batas waktu pelacakan koneksi idle:

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, Modifikasi batas waktu koneksi.
5. Modifikasi batas waktu pelacakan koneksi idle. Untuk informasi selengkapnya tentang opsi ini, 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 (5 hari). Default: 432.000 detik. 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 (3 menit). Default: 180 detik.
6. Pilih Simpan.

Untuk mengubah atribut antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [modify-network-interface-attribute](#) (AWS CLI)
- [Edit-EC2NetworkInterfaceAttribute](#) (AWS Tools for Windows PowerShell)

Menambahkan atau mengedit tanda

Tanda adalah metadata yang dapat Anda tambahkan ke antarmuka jaringan. Tanda bersifat privat dan hanya dapat dilihat oleh akun Anda. Setiap tanda terdiri dari kunci dan nilai opsional. Untuk informasi selengkapnya tentang cara memberikan tanda, lihat [Tandai sumber daya Amazon EC2 Anda](#).

Untuk menambah atau mengedit tanda untuk antarmuka jaringan menggunakan konsol

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. Di tab Tanda, pilih Kelola tanda.
5. Untuk setiap tanda yang akan dibuat, pilih Tambahkan tanda baru dan masukkan kunci dan nilai opsional. Setelah selesai, pilih Simpan.

Untuk menambah atau mengedit tanda untuk antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [create-tags](#) (AWS CLI)
- [New-EC2Tag](#) (AWS Tools for Windows PowerShell)

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](#).

Untuk menghapus antarmuka jaringan menggunakan konsol

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.

Untuk menghapus antarmuka jaringan menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [delete-network-interface](#) (AWS CLI)
- [Remove-EC2NetworkInterface](#) (AWS Tools for Windows PowerShell)

Praktik terbaik untuk mengonfigurasi antarmuka jaringan

- 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 memindahkan antarmuka jaringan sekunder dari satu instans ke instans lainnya, jika instans berada di Zona Ketersediaan dan VPC yang sama, tetapi dalam subnet yang berbeda.
- Saat meluncurkan instans menggunakan CLI, API, atau SDK, Anda dapat menentukan antarmuka jaringan primer dan antarmuka jaringan tambahan.
- 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 memasang antarmuka jaringan lain ke sebuah instans (misalnya, konfigurasi tim NIC) untuk menambah atau menggandakan bandwidth jaringan ke atau dari instans dual-homed.
- Jika Anda memasang dua atau lebih antarmuka jaringan dari subnet yang sama ke sebuah instans, Anda mungkin mengalami masalah jaringan seperti perutean asimetris. Jika memungkinkan, gunakan alamat IPv4 privat sekunder pada antarmuka jaringan primer sebagai gantinya.
- Instans Windows - Jika Anda menggunakan beberapa antarmuka jaringan, Anda harus mengkonfigurasi antarmuka jaringan untuk menggunakan perutean statis.

Mengonfigurasi antarmuka jaringan Anda menggunakan ec2-net-utils untuk Amazon Linux 2

Note

Untuk AL2023, `amazon-ec2-net-utils` paket menghasilkan konfigurasi khusus antarmuka di direktori `/run/systemd/network` Untuk informasi selengkapnya, lihat [Layanan jaringan](#) di Panduan Pengguna Amazon Linux 2023.

Amazon Linux 2 AMI mungkin berisi skrip tambahan yang diinstal oleh AWS, yang dikenal sebagai `ec2-net-utils`. Skrip ini secara opsional mengotomatiskan konfigurasi antarmuka jaringan Anda. Skrip ini hanya tersedia untuk Amazon Linux 2.

Gunakan perintah berikut untuk menginstal paket di Amazon Linux 2 jika belum diinstal, atau perbarui jika diinstal dan pembaruan tambahan tersedia:

```
$ yum install ec2-net-utils
```

Komponen berikut adalah bagian dari `ec2-net-utils`:

aturan udev (`/etc/udev/rules.d`)

Mengidentifikasi antarmuka jaringan saat dipasang, dilepaskan, atau dipasang kembali ke instans yang sedang berjalan, dan memastikan bahwa skrip hotplug berjalan (`53-ec2-network-interfaces.rules`). Memetakan alamat MAC ke nama perangkat (`75-persistent-net-generator.rules`, yang menghasilkan `70-persistent-net.rules`).

skrip hotplug

Menghasilkan file konfigurasi antarmuka yang cocok untuk digunakan dengan DHCP (`/etc/sysconfig/network-scripts/ifcfg-eth N`). Juga menghasilkan file konfigurasi rute (`/etc/sysconfig/network-scripts/route-eth N`).

Skrip DHCP

Setiap kali antarmuka jaringan menerima sewa DHCP baru, skrip ini menanyakan metadata instans untuk alamat IP Elastis. Untuk setiap alamat IP Elastis, ini menambahkan aturan ke basis data kebijakan perutean untuk memastikan bahwa lalu lintas ke luar dari alamat tersebut menggunakan antarmuka jaringan yang benar. Itu juga menambahkan setiap alamat IP privat ke antarmuka jaringan sebagai alamat sekunder.

ec2ifup ethN (/usr/sbin/)

Memperpanjang fungsi standar ifup. Setelah skrip ini menulis ulang file konfigurasi `ifcfg-ethN` dan `route-ethN`, ia menjalankan ifup.

ec2ifdown ethN (/usr/sbin/)

Memperpanjang fungsi standar ifdown. Setelah skrip ini menghapus semua aturan untuk antarmuka jaringan dari basis data kebijakan perutean, skrip akan menjalankan ifdown.

ec2ifscan (/usr/sbin/)

Memeriksa antarmuka jaringan yang belum dikonfigurasi dan mengonfigurasinya.

Skrip ini tidak tersedia pada rilis awal `ec2-net-utils`.

Untuk membuat daftar file konfigurasi yang dibuat oleh `ec2-net-utils`, gunakan perintah berikut:

```
$ ls -l /etc/sysconfig/network-scripts/*-eth?
```

Untuk menonaktifkan otomatisasi, Anda dapat menambahkan `EC2SYNC=no` ke file `ifcfg-eth N` yang sesuai. Misalnya, gunakan perintah berikut untuk menonaktifkan otomatisasi untuk antarmuka `eth1`:

```
$ sed -i -e 's/^EC2SYNC=yes/EC2SYNC=no/' /etc/sysconfig/network-scripts/ifcfg-eth1
```

Untuk menonaktifkan otomatisasi sepenuhnya, Anda dapat menghapus paket menggunakan perintah berikut:

```
$ yum remove ec2-net-utils
```

Skenario untuk antarmuka jaringan

Melampirkan beberapa antarmuka jaringan ke sebuah instans berguna ketika Anda ingin:

- Membuat jaringan manajemen.
- Menggunakan peralatan jaringan dan keamanan di Cloud Privat Virtual (VPC) Anda.
- Membuat instans dual-homed dengan beban kerja / peran pada subnet yang berbeda.
- Membuat solusi anggaran rendah dan ketersediaan tinggi.

Buat jaringan manajemen

Skenario ini menjelaskan bagaimana Anda dapat membuat jaringan manajemen dengan antarmuka jaringan, mengingat kriteria dan pengaturan berikut (gambar berikut).

Kriteria

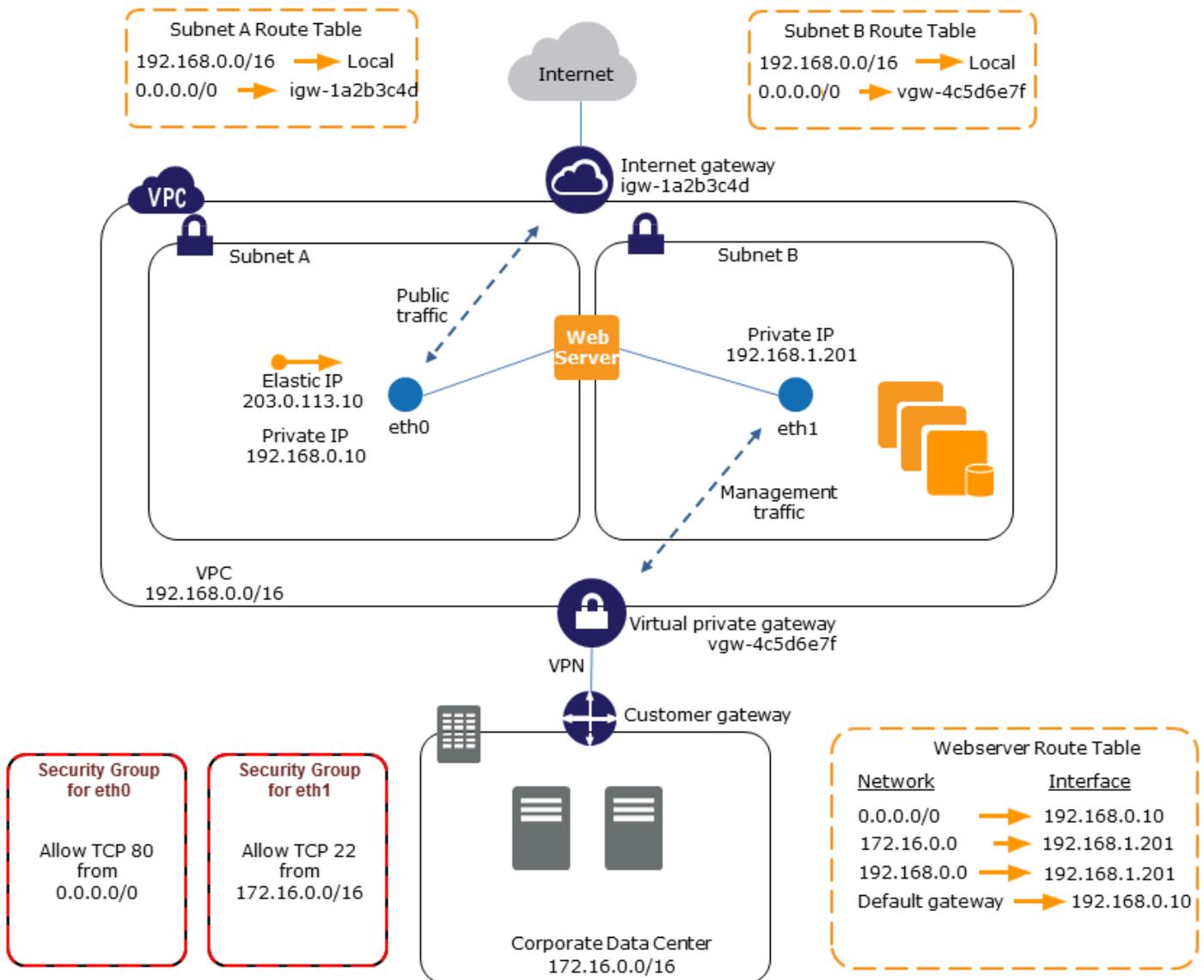
- Antarmuka jaringan primer pada instans (eth0) menangani lalu lintas publik.
- Antarmuka jaringan sekunder pada instans (eth1) menangani lalu lintas manajemen backend. Ini terhubung ke subnet terpisah yang memiliki kontrol akses yang lebih ketat, dan terletak di Zona Ketersediaan (AZ) yang sama dengan antarmuka jaringan primer.

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 privat dalam AZ yang sama dengan antarmuka jaringan primer.
 - Virtual private gateway.

Note

Untuk memastikan kemampuan failover, pertimbangkan untuk menggunakan IPv4 privat sekunder untuk lalu lintas masuk pada antarmuka jaringan. Jika terjadi kegagalan instans, Anda dapat memindahkan antarmuka dan/atau alamat IPv4 privat sekunder ke instans yang berjaga.



Gunakan peralatan jaringan dan keamanan di VPC Anda

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 memasang antarmuka jaringan sekunder ke instans yang menjalankan tipe aplikasi ini dan mengonfigurasi antarmuka tambahan dengan alamat IP publik dan privatnya sendiri, grup keamanan, dan pemeriksaan sumber/tujuan.

Membuat instans dual-homed dengan beban kerja / peran pada 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.

Membuat instans dual-homed dengan beban kerja/peran pada VPC yang berbeda dalam akun yang sama

Anda dapat meluncurkan instans EC2 dalam satu VPC dan melampirkan ENI sekunder dari VPC lain (tetapi di Zona Ketersediaan yang sama) ke instans. Ini memungkinkan Anda membuat instans multi-homed di seluruh VPC dengan konfigurasi jaringan dan keamanan yang berbeda. Anda tidak dapat membuat instance multi-homed di seluruh VPC di berbagai akun. 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 diintegrasikan 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.

Buat solusi ketersediaan tinggi anggaran rendah

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 guna 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.

Antarmuka jaringan yang dikelola pemohon

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

- Anda dapat melihat antarmuka jaringan yang dikelola pemohon di akun Anda. Anda dapat menambah atau menghapus tanda, tetapi Anda tidak dapat mengubah properti lain dari antarmuka jaringan yang dikelola pemohon.
- Anda tidak dapat melepaskan antarmuka jaringan yang dikelola pemohon.
- Saat Anda menghapus sumber daya yang terkait dengan antarmuka jaringan yang dikelola pemohon, antarmuka jaringan Layanan AWS terlepas dan menghapusnya. Jika layanan melepaskan antarmuka jaringan tetapi tidak menghapusnya, Anda dapat menghapus antarmuka jaringan yang terpisah.

Untuk melihat antarmuka jaringan yang dikelola pemohon menggunakan konsol

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.

Untuk melihat antarmuka jaringan yang dikelola pemohon menggunakan AWS CLI

Gunakan perintah [describe-network-interfaces](#) sebagai berikut.

```
aws ec2 describe-network-interfaces --filters Name=requester-managed,Values=true
```

Berikut ini adalah contoh output yang menunjukkan bidang kunci yang dapat Anda gunakan untuk menentukan tujuan antarmuka jaringan: `Description` dan `InterfaceType`.

```
{
  ...
  "Description": "VPC Endpoint Interface vpce-089f2123488812123",
  ...
  "InterfaceType": "vpc_endpoint",
  ...
  "NetworkInterfaceId": "eni-0d11e3ccd2c0e6c57",
  ...
  "RequesterId": "727180483921",
  "RequesterManaged": true,
  ...
}
```

Untuk melihat antarmuka jaringan yang dikelola pemohon menggunakan Alat untuk Windows PowerShell

Gunakan [Get-EC2NetworkInterface](#) cmdlet sebagai berikut.

```
Get-EC2NetworkInterface -Filter @{ Name="requester-managed"; Values="true" }
```

Berikut ini adalah contoh output yang menunjukkan bidang kunci yang dapat Anda gunakan untuk menentukan tujuan antarmuka jaringan: `Description` dan `InterfaceType`.

```
Description          : VPC Endpoint Interface vpce-089f2123488812123
...
InterfaceType        : vpc_endpoint
...
NetworkInterfaceId  : eni-0d11e3ccd2c0e6c57
...
RequesterId          : 727180483921
RequesterManaged    : True
...
```

Menetapkan prefiks ke 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 [Pengalaman IP instans Amazon EC2](#).

Pilihan penetapan berikut tersedia:

- Penugasan otomatis — AWS memilih awalan dari blok CIDR IPv4 atau IPv6 subnet VPC Anda dan menentukannya ke antarmuka jaringan Anda.
- Penugasan Manual - Anda menentukan awalan dari blok CIDR IPv4 atau IPv6 subnet VPC Anda, dan AWS memverifikasi bahwa awalan belum ditetapkan ke sumber daya lain sebelum menentukannya 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

- [Dasar-dasar untuk menetapkan prefiks](#)
- [Pertimbangan dan batasan untuk prefiks](#)
- [Bekerja dengan prefiks](#)

Dasar-dasar untuk menetapkan prefiks

- 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 dan batasan untuk prefiks

Pertimbangkan hal berikut ini saat Anda menggunakan prefiks:

- Antarmuka jaringan dengan awalan didukung dengan [instance yang dibangun di atas](#) Sistem Nitro. AWS
- Prefiks untuk antarmuka jaringan terbatas pada alamat IPv6 dan alamat IPv4 privat.
- 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 per antarmuka jaringan per tipe instans](#).

- Prefiks disertakan dalam pemeriksaan sumber/tujuan.

Bekerja dengan prefiks

Anda dapat menggunakan prefiks dengan antarmuka jaringan Anda sebagai berikut.

Tugas

- [Tetapkan prefiks selama pembuatan antarmuka jaringan](#)
- [Tetapkan prefiks ke antarmuka jaringan yang ada](#)
- [Konfigurasi sistem operasi Anda untuk antarmuka jaringan dengan prefiks](#)
- [Melihat prefiks yang ditetapkan ke antarmuka jaringan Anda](#)
- [Hapus prefiks dari antarmuka jaringan Anda](#)

Tetapkan prefiks selama pembuatan antarmuka jaringan

Jika Anda menggunakan opsi penugasan otomatis, Anda dapat memesan blok alamat IP di subnet Anda. AWS memilih awalan dari blok ini. Untuk informasi selengkapnya, lihat [Reservasi CIDR Subnet](#) di Panduan Pengguna Amazon VPC.

Setelah Anda membuat antarmuka jaringan, gunakan AWS CLI perintah [attach-network-interface untuk melampirkan antarmuka](#) jaringan ke instance Anda. Anda harus mengonfigurasi sistem operasi Anda untuk bekerja dengan antarmuka jaringan dengan prefiks. Untuk informasi selengkapnya, lihat [Konfigurasi sistem operasi Anda untuk antarmuka jaringan dengan prefiks](#).

Tugas

- [Tetapkan prefiks otomatis selama pembuatan antarmuka jaringan](#)
- [Tetapkan prefiks tertentu selama pembuatan antarmuka jaringan](#)

Tetapkan prefiks otomatis selama pembuatan antarmuka jaringan

Anda dapat menetapkan prefiks otomatis selama pembuatan antarmuka jaringan menggunakan salah satu metode berikut.

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, lalu pilih Buat antarmuka jaringan.
3. Tentukan deskripsi untuk antarmuka jaringan, pilih subnet untuk membuat antarmuka jaringan, dan konfigurasi alamat IPv4 dan IPv6 privat.
4. Perluas pengaturan lanjutan dan lakukan hal berikut ini:
 - a. Untuk secara otomatis menetapkan prefiks IPv4, untuk delegasi prefiks IPv4, pilih Menetapkan secara otomatis. Kemudian untuk Jumlah prefiks IPv4, tentukan jumlah prefiks yang akan ditetapkan.
 - b. Untuk secara otomatis menetapkan prefiks IPv6, untuk delegasi prefiks IPv6, pilih Menetapkan secara otomatis. Kemudian untuk Jumlah prefiks IPv6, tentukan jumlah prefiks yang akan ditetapkan.

Note

Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

5. Pilih grup keamanan yang akan dikaitkan dengan antarmuka jaringan dan tetapkan tanda sumber daya jika diperlukan.
6. Pilih Buat antarmuka jaringan.

AWS CLI

Untuk menetapkan prefiks IPv4 otomatis selama pembuatan antarmuka jaringan

Gunakan [create-network-interface](#) perintah dan atur `--ipv4-prefix-count` ke jumlah awalan yang AWS ingin Anda tetapkan. Dalam contoh berikut, AWS berikan 1 awalan.

```
$ C:\> aws ec2 create-network-interface \  
--subnet-id subnet-047cfed18eEXAMPLE \  
--description "IPv4 automatic example" \  
--ipv4-prefix-count 1
```

Contoh Output

```
{
  "NetworkInterface": {
    "AvailabilityZone": "us-west-2a",
    "Description": "IPv4 automatic example",
    "Groups": [
      {
        "GroupName": "default",
        "GroupId": "sg-044c2de2c4EXAMPLE"
      }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [],
    "MacAddress": "02:98:65:dd:18:47",
    "NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
    "OwnerId": "123456789012",
    "PrivateIpAddress": "10.0.0.62",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateIpAddress": "10.0.0.62"
      }
    ],
    "Ipv4Prefixes": [
      {
        "Ipv4Prefix": "10.0.0.208/28"
      }
    ],
    "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
    "RequesterManaged": false,
    "SourceDestCheck": true,
    "Status": "pending",
    "SubnetId": "subnet-047cfed18eEXAMPLE",
    "TagSet": [],
    "VpcId": "vpc-0e12f52b21EXAMPLE"
  }
}
```

Untuk menetapkan prefiks IPv6 otomatis selama pembuatan antarmuka jaringan

Gunakan [create-network-interface](#) perintah dan atur `--ipv6-prefix-count` ke jumlah awalan yang AWS ingin Anda tetapkan. Dalam contoh berikut, AWS berikan 1 awalan.

```
$ C:\> aws ec2 create-network-interface \
```

```
--subnet-id subnet-047cfed18eEXAMPLE \  
--description "IPv6 automatic example" \  
--ipv6-prefix-count 1
```

Contoh Output

```
{  
  "NetworkInterface": {  
    "AvailabilityZone": "us-west-2a",  
    "Description": "IPv6 automatic example",  
    "Groups": [  
      {  
        "GroupName": "default",  
        "GroupId": "sg-044c2de2c4EXAMPLE"  
      }  
    ],  
    "InterfaceType": "interface",  
    "Ipv6Addresses": [],  
    "MacAddress": "02:bb:e4:31:fe:09",  
    "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",  
    "OwnerId": "123456789012",  
    "PrivateIpAddress": "10.0.0.73",  
    "PrivateIpAddresses": [  
      {  
        "Primary": true,  
        "PrivateIpAddress": "10.0.0.73"  
      }  
    ],  
    "Ipv6Prefixes": [  
      {  
        "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"  
      }  
    ],  
    "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",  
    "RequesterManaged": false,  
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-047cfed18eEXAMPLE",  
    "TagSet": [],  
    "VpcId": "vpc-0e12f52b21EXAMPLE"  
  }  
}
```

Tetapkan prefiks tertentu selama pembuatan antarmuka jaringan

Anda dapat menetapkan prefiks tertentu selama pembuatan antarmuka jaringan menggunakan salah satu metode berikut.

Console

Untuk menetapkan prefiks tertentu selama pembuatan antarmuka jaringan

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. Tentukan deskripsi untuk antarmuka jaringan, pilih subnet untuk membuat antarmuka jaringan, dan konfigurasi alamat IPv4 dan IPv6 privat.
4. Perluas pengaturan lanjutan dan lakukan hal berikut ini:
 - a. Untuk menetapkan prefiks IPv4 tertentu, untuk delegasi prefiks IPv4, pilih Kustom. Kemudian pilih Tambahkan prefiks baru dan masukkan prefiks yang akan digunakan.
 - b. Untuk menetapkan prefiks IPv6 tertentu, untuk delegasi prefiks IPv6, pilih Kustom. Kemudian pilih Tambahkan prefiks baru dan masukkan prefiks yang akan digunakan.

Note

Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

5. Pilih grup keamanan yang akan dikaitkan dengan antarmuka jaringan dan tetapkan tanda sumber daya jika diperlukan.
6. Pilih Buat antarmuka jaringan.

AWS CLI

Untuk menetapkan prefiks IPv4 tertentu selama pembuatan antarmuka jaringan

Gunakan [create-network-interface](#) perintah dan atur `--ipv4-prefixes` ke awalan. AWS memilih alamat IP dari kisaran ini. Dalam contoh berikut, prefiks CIDR adalah `10.0.0.208/28`.

```
$ C:\> aws ec2 create-network-interface \  
  --subnet-id subnet-047cfed18eEXAMPLE \  
  --description "IPv4 manual example" \  
  --ipv4-prefixes 10.0.0.208/28
```

```
--ipv4-prefixes Ipv4Prefix=10.0.0.208/28
```

Contoh output

```
{
  "NetworkInterface": {
    "AvailabilityZone": "us-west-2a",
    "Description": "IPv4 manual example",
    "Groups": [
      {
        "GroupName": "default",
        "GroupId": "sg-044c2de2c4EXAMPLE"
      }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [],
    "MacAddress": "02:98:65:dd:18:47",
    "NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
    "OwnerId": "123456789012",
    "PrivateIpAddress": "10.0.0.62",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateIpAddress": "10.0.0.62"
      }
    ],
    "Ipv4Prefixes": [
      {
        "Ipv4Prefix": "10.0.0.208/28"
      }
    ],
    "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
    "RequesterManaged": false,
    "SourceDestCheck": true,
    "Status": "pending",
    "SubnetId": "subnet-047cfed18eEXAMPLE",
    "TagSet": [],
    "VpcId": "vpc-0e12f52b21EXAMPLE"
  }
}
```

Untuk menetapkan prefiks IPv6 tertentu selama pembuatan antarmuka jaringan

Gunakan `create-network-interface` perintah dan atur `--ipv6-prefixes` ke awalan. AWS memilih alamat IP dari kisaran ini. Dalam contoh berikut, prefiks CIDR adalah `2600:1f13:fc2:a700:1768::/80`.

```
$ C:\> aws ec2 create-network-interface \  
  --subnet-id subnet-047cfed18eEXAMPLE \  
  --description "IPv6 manual example" \  
  --ipv6-prefixes Ipv6Prefix=2600:1f13:fc2:a700:1768::/80
```

Contoh output

```
{  
  "NetworkInterface": {  
    "AvailabilityZone": "us-west-2a",  
    "Description": "IPv6 automatic example",  
    "Groups": [  
      {  
        "GroupName": "default",  
        "GroupId": "sg-044c2de2c4EXAMPLE"  
      }  
    ],  
    "InterfaceType": "interface",  
    "Ipv6Addresses": [],  
    "MacAddress": "02:bb:e4:31:fe:09",  
    "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",  
    "OwnerId": "123456789012",  
    "PrivateIpAddress": "10.0.0.73",  
    "PrivateIpAddresses": [  
      {  
        "Primary": true,  
        "PrivateIpAddress": "10.0.0.73"  
      }  
    ],  
    "Ipv6Prefixes": [  
      {  
        "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"  
      }  
    ],  
    "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",  
    "RequesterManaged": false,  
    "SourceDestCheck": true,  
    "Status": "pending",  
    "SubnetId": "subnet-047cfed18eEXAMPLE",
```

```
    "TagSet": [],  
    "VpcId": "vpc-0e12f52b21EXAMPLE"  
  }  
}
```

Tetapkan prefiks ke antarmuka jaringan yang ada

Setelah Anda menetapkan awalan, gunakan [attach-network-interface](#) AWS CLI perintah untuk melampirkan antarmuka jaringan ke instance Anda. Anda harus mengonfigurasi sistem operasi Anda untuk bekerja dengan antarmuka jaringan dengan prefiks. Untuk informasi selengkapnya, lihat [Konfigurasi sistem operasi Anda untuk antarmuka jaringan dengan prefiks](#).

Tugas

- [Tetapkan prefiks otomatis ke antarmuka jaringan yang ada](#)
- [Tetapkan prefiks spesifik ke antarmuka jaringan yang ada](#)

Tetapkan prefiks otomatis ke antarmuka jaringan yang ada

Anda dapat menetapkan prefiks otomatis ke antarmuka jaringan yang ada menggunakan salah satu metode berikut.

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 secara otomatis menetapkan prefiks IPv4, untuk delegasi prefiks IPv4, pilih Menetapkan secara otomatis. Kemudian untuk Jumlah prefiks IPv4, tentukan jumlah prefiks yang akan ditetapkan.
5. Untuk secara otomatis menetapkan prefiks IPv6, untuk delegasi prefiks IPv6, pilih Menetapkan secara otomatis. Kemudian untuk Jumlah prefiks IPv6, tentukan jumlah prefiks yang akan ditetapkan.

Note

Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

6. Pilih Simpan.**AWS CLI**

Anda dapat menggunakan perintah [assign-ipv6-addresses](#) untuk menetapkan prefiks IPv6 dan perintah [assign-private-ip-addresses](#) untuk menetapkan prefiks IPv4 ke antarmuka jaringan yang ada.

Untuk menetapkan prefiks IPv4 otomatis ke antarmuka jaringan yang ada

Gunakan [assign-private-ip-addresses](#) perintah dan atur `--ipv4-prefix-count` ke jumlah awalan yang AWS ingin Anda tetapkan. Dalam contoh berikut, AWS menetapkan awalan 1 IPv4.

```
$ C:\> aws ec2 assign-private-ip-addresses \  
--network-interface-id eni-081fbb4095EXAMPLE \  
--ipv4-prefix-count 1
```

Contoh Output

```
{  
  "NetworkInterfaceId": "eni-081fbb4095EXAMPLE",  
  "AssignedIpv4Prefixes": [  
    {  
      "Ipv4Prefix": "10.0.0.176/28"  
    }  
  ]  
}
```

Untuk menetapkan prefiks IPv6 otomatis ke antarmuka jaringan yang ada

Gunakan perintah [assign-ipv6-address](#) dan atur `--ipv6-prefix-count` ke jumlah awalan yang ingin Anda tetapkan. AWS Dalam contoh berikut, AWS menetapkan awalan 1 IPv6.

```
$ C:\> aws ec2 assign-ipv6-addresses \  

```

```
--network-interface-id eni-00d577338cEXAMPLE \  
--ipv6-prefix-count 1
```

Contoh Output

```
{  
  "AssignedIpv6Prefixes": [  
    "2600:1f13:fc2:a700:18bb::/80"  
  ],  
  "NetworkInterfaceId": "eni-00d577338cEXAMPLE"  
}
```

Tetapkan prefiks spesifik ke antarmuka jaringan yang ada

Anda dapat menetapkan prefiks tertentu ke antarmuka jaringan yang ada menggunakan salah satu metode berikut.

Console

Untuk menetapkan prefiks tertentu 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 menetapkan prefiks IPv4 tertentu, untuk delegasi prefiks IPv4, pilih Kustom. Kemudian pilih Tambahkan prefiks baru dan masukkan prefiks yang akan digunakan.
5. Untuk menetapkan prefiks IPv6 tertentu, untuk delegasi prefiks IPv6, pilih Kustom. Kemudian pilih Tambahkan prefiks baru dan masukkan prefiks yang akan digunakan.

Note

Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

6. Pilih Simpan.

AWS CLI

Tetapkan prefiks IPv4 tertentu ke antarmuka jaringan yang ada

Gunakan [assign-private-ip-addresses](#) perintah dan atur `--ipv4-prefixes` ke awalan. AWS memilih alamat IPv4 dari kisaran ini. Dalam contoh berikut, prefiks CIDR adalah `10.0.0.208/28`.

```
$ C:\> aws ec2 assign-private-ip-addresses \  
--network-interface-id eni-081fbb4095EXAMPLE \  
--ipv4-prefixes 10.0.0.208/28
```

Contoh output

```
{  
  "NetworkInterfaceId": "eni-081fbb4095EXAMPLE",  
  "AssignedIpv4Prefixes": [  
    {  
      "Ipv4Prefix": "10.0.0.208/28"  
    }  
  ]  
}
```

Tetapkan prefiks IPv6 tertentu ke antarmuka jaringan yang ada

Gunakan perintah [assign-ipv6-address](#) dan atur ke awalan. `--ipv6-prefixes` AWS memilih alamat IPv6 dari kisaran ini. Dalam contoh berikut, prefiks CIDR adalah `2600:1f13:fc2:a700:18bb::/80`.

```
$ C:\> aws ec2 assign-ipv6-addresses \  
--network-interface-id eni-00d577338cEXAMPLE \  
--ipv6-prefixes 2600:1f13:fc2:a700:18bb::/80
```

Contoh output

```
{  
  "NetworkInterfaceId": "eni-00d577338cEXAMPLE",  
  "AssignedIpv6Prefixes": [  
    {  
      "Ipv6Prefix": "2600:1f13:fc2:a700:18bb::/80"  
    }  
  ]  
}
```

Konfigurasi sistem operasi Anda untuk antarmuka jaringan dengan prefiks

Amazon Linux AMI mungkin berisi skrip tambahan yang diinstal oleh AWS, yang dikenal sebagai `ec2-net-utils`. Skrip ini secara opsional mengotomatiskan konfigurasi antarmuka jaringan Anda. Mereka hanya tersedia untuk Amazon Linux.

Jika Anda tidak menggunakan Amazon Linux, Anda dapat menggunakan Container Network Interface (CNI) untuk plug-in Kubernetes, atau `dockerd` jika Anda menggunakan Docker untuk mengelola container Anda.

Melihat prefiks yang ditetapkan ke antarmuka jaringan Anda

Anda dapat melihat prefiks yang ditetapkan ke antarmuka jaringan Anda menggunakan salah satu metode berikut.

Console

Untuk melihat prefiks otomatis yang ditetapkan 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 untuk melihat prefiks dan pilih tab Detail.
4. Bidang Delegasi Prefiks IPv4 mencantumkan prefiks IPv4 yang ditetapkan, dan bidang Delegasi Prefiks IPv6 mencantumkan prefiks IPv6 yang ditetapkan.

AWS CLI

Anda dapat menggunakan [describe-network-interfaces](#) AWS CLI perintah untuk melihat awalan yang ditetapkan ke antarmuka jaringan Anda.

```
$ C:\> aws ec2 describe-network-interfaces
```

Contoh Output

```
{
  "NetworkInterfaces": [
    {
      "AvailabilityZone": "us-west-2a",
      "Description": "IPv4 automatic example",
      "Groups": [
```

```
        {
            "GroupName": "default",
            "GroupId": "sg-044c2de2c4EXAMPLE"
        }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [],
    "MacAddress": "02:98:65:dd:18:47",
    "NetworkInterfaceId": "eni-02b80b4668EXAMPLE",
    "OwnerId": "123456789012",
    "PrivateIpAddress": "10.0.0.62",
    "PrivateIpAddresses": [
        {
            "Primary": true,
            "PrivateIpAddress": "10.0.0.62"
        }
    ],
    "Ipv4Prefixes": [
        {
            "Ipv4Prefix": "10.0.0.208/28"
        }
    ],
    "Ipv6Prefixes": [],
    "RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
    "RequesterManaged": false,
    "SourceDestCheck": true,
    "Status": "available",
    "SubnetId": "subnet-05eef9fb78EXAMPLE",
    "TagSet": [],
    "VpcId": "vpc-0e12f52b2146bf252"
},
{
    "AvailabilityZone": "us-west-2a",
    "Description": "IPv6 automatic example",
    "Groups": [
        {
            "GroupName": "default",
            "GroupId": "sg-044c2de2c411c91b5"
        }
    ],
    "InterfaceType": "interface",
    "Ipv6Addresses": [],
    "MacAddress": "02:bb:e4:31:fe:09",
    "NetworkInterfaceId": "eni-006edbcfa4EXAMPLE",
```

```
"OwnerId": "123456789012",
"PrivateIpAddress": "10.0.0.73",
"PrivateIpAddresses": [
  {
    "Primary": true,
    "PrivateIpAddress": "10.0.0.73"
  }
],
"Ipv4Prefixes": [],
"Ipv6Prefixes": [
  {
    "Ipv6Prefix": "2600:1f13:fc2:a700:1768::/80"
  }
],
"RequesterId": "AIDAIV5AJI5LXF5XXDPC0",
"RequesterManaged": false,
"SourceDestCheck": true,
"Status": "available",
"SubnetId": "subnet-05eef9fb78EXAMPLE",
"TagSet": [],
"VpcId": "vpc-0e12f52b21EXAMPLE"
}
]
}
```

Hapus prefiks dari antarmuka jaringan Anda

Anda dapat menghapus prefiks dari antarmuka jaringan Anda menggunakan salah satu metode berikut.

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 untuk menghapus prefiks dan pilih Tindakan, Kelola prefiks.
4. Lakukan salah satu langkah berikut:
 - Untuk menghapus semua prefiks yang ditetapkan, untuk delegasi prefiks IPv4 dan delegasi prefiks IPv6, pilih Jangan tetapkan.

- Untuk menghapus prefiks tertentu yang ditetapkan, untuk delegasi prefiks IPv4 atau delegasi prefiks IPv6, pilih Kustom, lalu pilih Batalkan penetapan di sebelah prefiks yang akan dihapus.

 Note

Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

5. 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-addresses](#) dan atur `--ipv4-prefix` ke alamat yang ingin Anda hapus.

```
$ C:\> 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-addresses](#) dan atur `--ipv6-prefix` ke alamat yang ingin Anda hapus.

```
$ C:\> aws ec2 unassign-ipv6-addresses \  
--network-interface-id eni-00d577338cEXAMPLE \  
--ipv6-prefix 2600:1f13:fc2:a700:18bb::/80
```

Bandwidth jaringan instans Amazon EC2

Spesifikasi bandwidth instans berlaku untuk lalu lintas masuk dan keluar untuk instans. Misalnya, jika sebuah instans menentukan bandwidth hingga 10 Gbps, itu berarti ia memiliki bandwidth hingga 10 Gbps untuk lalu lintas masuk, dan hingga 10 Gbps untuk lalu lintas keluar. Bandwidth jaringan yang tersedia untuk instans EC2 tergantung pada beberapa faktor, sebagai berikut.

Lalu lintas multi-aliran

Bandwidth untuk lalu lintas multi-alur agregat yang tersedia untuk suatu instans tergantung pada tujuan lalu lintas.

- Di Wilayah — Lalu lintas dapat memanfaatkan bandwidth jaringan penuh yang tersedia untuk instans.
- Ke Wilayah lain, gateway internet, Direct Connect, atau gateway lokal (LGW) — Lalu lintas dapat memanfaatkan hingga 50% dari bandwidth jaringan yang tersedia untuk instans generasi saat ini dengan minimal 32 vCPU. Bandwidth untuk instans generasi saat ini dengan kurang dari 32 vCPU dibatasi hingga 5 Gbps.

Lalu lintas alur tunggal

Bandwidth dasar untuk lalu lintas aliran tunggal dibatasi hingga 5 Gbps ketika instans tidak berada dalam [grup penempatan klaster](#) yang sama. 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).
- Konfigurasi ENA Ekspres untuk instans yang memenuhi syarat dalam subnet yang sama untuk mencapai hingga 25 Gbps di antara instans tersebut.

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 instans `m5.16xlarge` memiliki 64 vCPU, sehingga lalu lintas ke instans lain di Wilayah dapat memanfaatkan bandwidth penuh yang tersedia (20 Gbps). Namun, lalu lintas ke instans lain di Wilayah yang berbeda hanya dapat memanfaatkan 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 kredit I/O 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 instans menerima jumlah maksimum kredit I/O jaringan saat peluncuran. Jika instans menghabiskan kredit I/O jaringannya, ia kembali ke bandwidth baseline. Sebuah instans yang berjalan menghasilkan kredit I/O jaringan setiap kali menggunakan bandwidth jaringan lebih sedikit daripada bandwidth dasarnya. Instans yang dihentikan tidak mendapatkan kredit I/O jaringan. Lonjakan instans adalah upaya terbaik, bahkan ketika instans memiliki kredit yang tersedia, karena lonjakan bandwidth adalah sumber daya bersama.

Ada bucket kredit I/O jaringan terpisah untuk lalu lintas masuk dan keluar.

Performa jaringan dasar dan lonjakan

Panduan Jenis Instans Amazon EC2 menjelaskan performa 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 - Komputasi kinerja tinggi](#)
- [Spesifikasi jaringan — Generasi sebelumnya](#)

Untuk melihat kinerja jaringan menggunakan AWS CLI

Anda dapat menggunakan AWS CLI 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]" --output table
```

```
-----
|           DescribeInstanceTypes           |
+-----+-----+-----+
| c5.4xlarge | Up to 10 Gigabit | 5.0 |
| c5.xlarge  | Up to 10 Gigabit | 1.25 |
| c5.12xlarge | 12 Gigabit       | 12.0 |
| c5.24xlarge | 25 Gigabit       | 25.0 |
| c5.metal   | 25 Gigabit       | 25.0 |
| c5.9xlarge | 12 Gigabit       | 12.0 |
| c5.2xlarge | Up to 10 Gigabit | 2.5 |
| c5.large   | Up to 10 Gigabit | 0.75 |
| c5.18xlarge | 25 Gigabit       | 25.0 |
+-----+-----+-----+
```

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 pendek dalam permintaan untuk sumber daya jaringan (dikenal sebagai microburst), tetapi CloudWatch metriknya tidak cukup terperinci untuk mencerminkan lonjakan mikrodetik ini.

Pelajari selengkapnya

- [Metrik instans](#)
- [Metrik performa jaringan](#)

Jaringan yang disempurnakan di Amazon EC2

Jaringan yang ditingkatkan menggunakan virtualisasi I/O root tunggal (SR-IOV) untuk menyediakan kapabilitas jaringan berperforma tinggi pada [tipe instans yang didukung](#). SR-IOV adalah metode virtualisasi perangkat yang memberikan performa I/O lebih tinggi dan penggunaan CPU lebih rendah jika dibandingkan dengan antarmuka jaringan divirtualisasi secara tradisional. Jaringan yang ditingkatkan memberikan bandwidth yang lebih tinggi, performa paket per detik yang lebih tinggi (PPS), dan latensi antar-instans lebih rendah yang konsisten. 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](#).

Daftar Isi

- [Dukungan jaringan yang ditingkatkan](#)
- [Aktifkan jaringan yang disempurnakan dengan Elastic Network Adapter \(ENA\) pada instans EC2](#)
- [Tingkatkan kinerja jaringan dengan ENA Express pada instans EC2 Anda](#)
- [Aktifkan jaringan yang disempurnakan dengan antarmuka Intel 82599 VF pada instans EC2](#)
- [Memantau performa jaringan untuk instans EC2 Anda](#)
- [Memecahkan Masalah Adaptor Jaringan Elastis di Linux](#)
- [Memecahkan masalah driver Windows Adaptor Jaringan Elastis](#)
- [Meningkatkan latensi jaringan untuk instans Amazon EC2 berbasis Linux](#)
- [Pertimbangan sistem nitro untuk penyetelan kinerja](#)
- [Optimalkan kinerja jaringan pada instance Windows](#)

Dukungan jaringan yang ditingkatkan

Semua tipe instans generasi saat ini mendukung jaringan yang ditingkatkan, kecuali untuk instans T2.

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 [instance yang dibangun di atas Sistem AWS Nitro](#) menggunakan ENA untuk meningkatkan jaringan. Selain itu, tipe instans Xen berikut mendukung ENA: H1, G3, m4.16xlarge, P2, P3, P3dn, dan R4.

Untuk informasi selengkapnya, lihat [Aktifkan jaringan yang disempurnakan dengan Elastic Network Adapter \(ENA\) pada instans EC2](#).

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 [Aktifkan jaringan yang disempurnakan dengan antarmuka Intel 82599 VF pada instans EC2](#).

Aktifkan jaringan yang disempurnakan dengan Elastic Network Adapter (ENA) pada instans EC2

Amazon EC2 memberikan kemampuan jaringan yang ditingkatkan melalui Adaptor Jaringan Elastis (ENA). Untuk menggunakan jaringan yang ditingkatkan, Anda harus menginstal modul ENA yang diperlukan dan mengaktifkan dukungan ENA.

Daftar Isi

- [Persyaratan](#)
- [Performa jaringan yang ditingkatkan](#)
- [AMI Linux dengan modul yang diperlukan](#)
- [Menguji apakah jaringan yang ditingkatkan diaktifkan](#)
- [Mengaktifkan jaringan yang ditingkatkan pada instans Anda](#)
- [Catatan rilis driver](#)

Persyaratan

Untuk mempersiapkan jaringan yang ditingkatkan menggunakan ENA, siapkan instans Anda sebagai berikut:

- Luncurkan [instance yang dibangun di atas Sistem AWS Nitro](#).
- 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 `enaSupport`, 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 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 itu memiliki pembaruan dukungan penandatanganan [kode SHA-2](#).
- Gunakan [AWS CloudShell](#) dari AWS Management Console, atau instal dan konfigurasi [AWS CLI](#) atau [AWS Tools for Windows PowerShell](#) di 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.

Performa jaringan yang ditingkatkan

Dokumentasi berikut memberikan ringkasan performa jaringan untuk tipe instans yang mendukung jaringan yang ditingkatkan ENA:

- [Spesifikasi jaringan untuk instans komputasi yang dipercepat](#)
- [Spesifikasi jaringan untuk menghitung instans yang dioptimalkan](#)
- [Spesifikasi jaringan untuk contoh tujuan umum](#)
- [Spesifikasi jaringan untuk instans komputasi berkinerja tinggi](#)
- [Spesifikasi jaringan untuk instance yang dioptimalkan memori](#)
- [Spesifikasi jaringan untuk instans penyimpanan yang dioptimalkan](#)

AMI Linux dengan modul yang diperlukan

AMI berikut mencakup modul ENA yang diperlukan dan telah mengaktifkan dukungan ENA:

- AL2023

- Amazon Linux 2
- Amazon Linux AMI 2018.03 dan setelahnya
- Ubuntu 14.04 atau setelahnya dengan kernel `linux-aws`

 Note

AWS Jenis instance berbasis Graviton 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 setelahnya
- Debian GNU/Linux 9 atau setelahnya

Untuk menguji apakah jaringan yang disempurnakan sudah diaktifkan, verifikasi bahwa ena modul diinstal pada instance Anda dan enaSupport atribut disetel. Jika demikian, maka perintah `ethtool -i ethn` harus menunjukkan bahwa modul sedang digunakan pada antarmuka jaringan.

Modul kernel (ena)

Untuk memverifikasi modul ena telah diinstal, gunakan perintah `modinfo` seperti yang ditunjukkan pada contoh berikut.

```
[ec2-user ~]$ modinfo ena
filename:      /lib/modules/4.14.33-59.37.amzn2.x86_64/kernel/drivers/amazon/net/ena/
ena.ko
version:      1.5.0g
license:      GPL
description:  Elastic Network Adapter (ENA)
author:       Amazon.com, Inc. or its affiliates
srcversion:   692C7C68B8A9001CB3F31D0
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:
retpoline:    Y
intree:       Y
```

```
name:      ena
...
```

Dalam contoh Amazon Linux, ena modul diinstal.

```
ubuntu:~$ modinfo ena
ERROR: modinfo: could not find module ena
```

Dalam contoh Ubuntu, modul tidak diinstal, jadi Anda harus menginstalnya terlebih dahulu. Untuk informasi selengkapnya, lihat [Ubuntu](#).

Menguji apakah jaringan yang ditingkatkan diaktifkan

Anda dapat menguji apakah jaringan yang disempurnakan diaktifkan di instans Anda atau AMI Anda.

Atribut contoh

Untuk memeriksa apakah sebuah instans memiliki set atribut `enaSupport` jaringan yang ditingkatkan, gunakan salah satu dari perintah berikut. Jika atributnya ditetapkan, responsnya adalah benar.

- [describe-instances](#) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instances --instance-ids instance_id --query
"Reservations[].Instances[].EnaSupport"
```

- [Get-EC2Instance](#) (Alat untuk Windows PowerShell)

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

Atribut gambar

Untuk memeriksa apakah AMI memiliki set atribut `enaSupport` jaringan yang ditingkatkan, gunakan salah satu dari perintah berikut. Jika atributnya ditetapkan, responsnya adalah true.

- [describe-images](#) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-images --image-id ami_id --query "Images[].EnaSupport"
```

- [Get-EC2Image](#) (Alat untuk Windows PowerShell)

```
(Get-EC2Image -ImageId ami_id).EnaSupport
```

Driver antarmuka jaringan Linux

Gunakan perintah berikut untuk memverifikasi apakah modul ena sedang digunakan pada antarmuka tertentu, menggantikan nama antarmuka yang ingin Anda periksa. Jika Anda menggunakan antarmuka tunggal (default), ini eth0. Jika sistem operasi mendukung [nama jaringan yang dapat diprediksi](#), ini bisa menjadi nama seperti ens5.

Dalam contoh berikut, modul ena 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 ena 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

Amazon Linux 2 dan versi AMI Amazon Linux terbaru mencakup modul yang diperlukan untuk jaringan yang ditingkatkan dengan ENA terinstal dan telah mengaktifkan dukungan ENA. 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](#).

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 di Amazon Linux AMI

1. Connect ke instans Anda.
2. Dari instans, jalankan perintah berikut untuk memperbarui instans Anda dengan modul kernel dan kernel terbaru, termasuk ena:

```
[ec2-user ~]$ sudo yum update
```

3. Dari komputer lokal Anda, reboot instans Anda menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [reboot-instances](#) (AWS CLI), [Restart-EC2Instance](#) (AWS Tools for Windows PowerShell).
4. Hubungkan lagi ke instans Anda dan verifikasi bahwa modul ena telah diinstal dan pada versi minimum yang disarankan menggunakan perintah `modinfo ena` dari [Menguji apakah jaringan yang ditingkatkan diaktifkan](#).
5. [Instans yang didukung EBS] Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu dari perintah berikut: [stop-instances](#) (AWS CLI), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

[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 Amazon Linux AMI \(instans yang didukung penyimpanan instans\)](#).

6. 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 instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#) (Alat untuk Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

7. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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.
8. Dari komputer lokal Anda, mulai instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
9. Hubungkan ke instans Anda dan verifikasi bahwa modul `ena` diinstal dan dimuat di antarmuka jaringan Anda menggunakan perintah `ethtool -i ethn` dari [Menguji apakah jaringan yang ditingkatkan diaktifkan](#).

Jika Anda tidak dapat terhubung ke instans Anda setelah mengaktifkan jaringan yang ditingkatkan, lihat [Memecahkan Masalah Adaptor Jaringan Elastis di Linux](#).

Untuk mengaktifkan jaringan yang ditingkatkan di Amazon Linux AMI (instans yang didukung penyimpanan instans)

Ikuti prosedur sebelumnya hingga langkah tempat Anda menghentikan instans. Buat AMI baru seperti yang dijelaskan di [Buat AMI Linux yang didukung penyimpanan instans](#), 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 ...
```

Ubuntu

AMI HVM Ubuntu terbaru mencakup modul yang diperlukan untuk jaringan yang ditingkatkan dengan ENA terinstal dan telah mengaktifkan dukungan ENA. 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 [AWS Support](#).

Untuk menginstal paket kernel **linux-aws** (Ubuntu Trusty 14.04)

1. Connect 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. [Instans yang didukung EBS] Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu dari perintah berikut: [stop-instances](#) (AWS CLI), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

[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 instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#) (Alat untuk Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

5. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

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 AMI Linux yang didukung penyimpanan instans](#), 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 versi terbaru untuk Red Hat Enterprise Linux, SUSE Linux Enterprise Server, dan CentOS menyertakan modul yang diperlukan untuk jaringan yang ditingkatkan dengan ENA dan telah mengaktifkan dukungan ENA. 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 ena modul pada instance Anda dari GitHub at<https://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. Lakukan kompilasi dan instal modul ena pada instans Anda. Langkah-langkah ini bergantung pada distribusi Linux. Untuk informasi selengkapnya tentang mengompilasi modul 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 perintah `sudo depmod` untuk memperbarui dependensi modul.
5. Perbarui `initramfs` pada instans Anda untuk memastikan bahwa modul 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 versi `systemd` atau `udev` pada sistem berbasis RPM 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 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`.

```
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. [Instans yang didukung EBS] Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu dari perintah berikut: [stop-instances](#) (AWS CLI), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

[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 instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#) (Alat untuk Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
```

- (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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.

- Dari komputer lokal Anda, mulai instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
- (Opsional) Hubungkan ke instans Anda dan verifikasi bahwa modul telah diinstal.

Jika Anda tidak dapat terhubung ke instans Anda setelah mengaktifkan jaringan yang ditingkatkan, lihat [Memecahkan Masalah Adaptor Jaringan Elastis 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 AMI Linux yang didukung penyimpanan instans](#), 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 (instans yang didukung EBS)

1. Ikuti langkah 1 dan 2 dalam [Ubuntu](#).
2. Instal paket `build-essential` untuk mengompilasi modul kernel dan paket `dkms`, sehingga modul `ena` Anda dibangun kembali setiap kali kernel Anda diperbarui.

```
ubuntu:~$ sudo apt-get install -y build-essential dkms
```

3. Kloning sumber untuk `ena` modul pada instance Anda dari GitHub at <https://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 modul ena pada instans Anda menggunakan DKMS.

Tambahkan modul ke DKMS.

```
ubuntu:~$ sudo dkms add -m amzn-drivers -v 1.0.0
```

Buat modul menggunakan perintah dkms.

```
ubuntu:~$ sudo dkms build -m amzn-drivers -v 1.0.0
```

Instal modul menggunakan dkms.

```
ubuntu:~$ sudo dkms install -m amzn-drivers -v 1.0.0
```

7. Buat kembali `initramfs` agar modul yang benar dimuat pada saat boot.

```
ubuntu:~$ sudo update-initramfs -u -k all
```

8. Pastikan modul ena diinstal menggunakan perintah `modinfo ena` dari [Menguji 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. Anda hanya dapat mengaktifkan atribut ini pada tipe instans yang didukung dan hanya jika driver ENA diinstal. Untuk informasi selengkapnya, lihat [Dukungan 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [stop-instances](#) (AWS CLI/AWS CloudShell), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
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 CloudShell)

```
aws ec2 describe-instances --instance-ids instance_id --query  
"Reservations[].Instances[].EnaSupport"
```

- [Get-EC2Instance](#) (Alat untuk Windows PowerShell)

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

- b. Untuk mengaktifkan dukungan ENA, jalankan salah satu dari perintah berikut ini:

- [modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -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 CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
```

- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $false
```

- 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI/AWS CloudShell), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance menggunakan AWS OpsWorks konsol sehingga status instance tetap sinkron.
7. Pada instans ini, validasi bahwa driver ENA diinstal dan diaktifkan sebagai berikut:
 - a. Klik kanan ikon jaringan dan pilih Buka Pusat Jaringan dan 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.

Catatan rilis driver

Driver Linux ENA

Untuk informasi tentang versi driver Linux ENA, lihat [Catatan rilis driver kernel ENA Linux](#).

Pengemudi Windows ENA

AMI Windows menyertakan driver Amazon ENA untuk mengaktifkan jaringan yang ditingkatkan.

Tabel berikut menunjukkan versi driver ENA yang sesuai untuk diunduh untuk setiap versi Windows Server.

Versi Windows Server	Versi driver ENA
Windows Server 2022	2.4.0 dan versi yang lebih baru
Windows Server 2019	terbaru
Windows Server 2016	terbaru
Windows Server 2012 R2	2.6.0 dan sebelumnya
Windows Server 2012	2.6.0 dan sebelumnya
Windows Server 2008 R2	2.2.3 dan sebelumnya

Tabel berikut merangkum perubahan untuk setiap rilis.

Versi driver	Detail	Tanggal rilis
2.7.0	Fitur Baru <ul style="list-style-type: none"> Dihapus dukungan untuk Windows Server 2012 (Windows 8) dan Windows Server 2012 R2 (Windows 8.1). Versi sistem operasi ini telah mencapai akhir 	1 Mei 2024

Versi driver	Detail	Tanggal rilis
	<p>dukungan dari AWS. Penginstalan driver akan gagal pada Windows Server 2012 dan sebelumnya.</p> <ul style="list-style-type: none">• Menambahkan dukungan untuk membongkar perhitungan checksum IPv6 Tx ke perangkat.• Menambahkan dukungan Low Latency Queuing (LLQ) yang luas. Ini diaktifkan secara dinamis berdasarkan rekomendasi perangkat. Anda dapat mengganti pengaturan ini dengan kunci registri “WidellQ” yang baru.• Menambahkan pelaporan untuk penurunan paket yang dihasilkan dari Rx overrun, yang menunjukkan ruang yang tidak cukup di ring Rx untuk paket yang masuk.• Menambahkan dukungan untuk pemberitahuan konfigurasi suboptimal dari perangkat. Lihat ID peristiwa 59000 di penampil acara Windows. <p>Perbaikan Bug</p> <ul style="list-style-type: none">• Hindari reset perangkat yang tidak perlu yang disebabkan oleh paket Tx dengan header yang melebihi ukuran header Low Latency Queuing (LLQ) maksimum.	

Versi driver	Detail	Tanggal rilis
2.6.0	<p data-bbox="402 260 542 289">Fitur Baru</p> <ul data-bbox="402 344 1192 1310" style="list-style-type: none"><li data-bbox="402 365 1192 449">• Menambahkan metrik performa jaringan berikut untuk tipe instans yang mendukung ENA Ekspres.<ul data-bbox="435 491 922 911" style="list-style-type: none"><li data-bbox="435 512 695 541">• <code>ena_srd_mode</code><li data-bbox="435 596 753 625">• <code>ena_srd_tx_pkts</code><li data-bbox="435 680 922 709">• <code>ena_srd_eligible_tx_pkts</code><li data-bbox="435 764 753 793">• <code>ena_srd_rx_pkts</code><li data-bbox="435 848 1003 877">• <code>ena_srd_resource_utilization</code><li data-bbox="402 953 1192 1079">• Menambahkan metrik performa jaringan <code>contrack_allowance_available</code> untuk tipe instans berbasis Nitro.<li data-bbox="402 1142 1127 1226">• Menambahkan alasan reset adaptor baru karena deteksi kerusakan data RX.<li data-bbox="402 1268 997 1310">• Perbarui infrastruktur pencatatan driver. <p data-bbox="402 1415 613 1444">Perbaikan Bug</p> <ul data-bbox="402 1499 1159 1835" style="list-style-type: none"><li data-bbox="402 1520 1159 1646">• Mencegah reset adaptor jika terjadi kelaparan CPU menyebabkan pembaruan metrik performa jaringan gagal.<li data-bbox="402 1709 1127 1793">• Cegah deteksi palsu interupsi pada detak jantung perangkat.<li data-bbox="402 1814 418 1835">•	20 Juni 2023

Versi driver	Detail	Tanggal rilis
	<p>Memperbaiki skrip instalasi driver untuk mendukung operasi downgrade.</p> <ul style="list-style-type: none">• Memperbaiki statistik jumlah kesalahan penerimaan.	
2.5.0	<p>Pengumuman</p> <p>ENA Windows driver versi 2.5.0 telah dibatalkan karena kegagalan untuk menginisialisasi pada pengontrol domain Windows. Windows Client dan Windows Server tidak terpengaruh.</p>	17 Februari 2023

Versi driver	Detail	Tanggal rilis
2.4.0	<p data-bbox="402 226 542 260">Fitur Baru</p> <ul data-bbox="402 310 1214 646" style="list-style-type: none"><li data-bbox="402 310 1214 373">• Menambahkan dukungan untuk Windows Server 2022.<li data-bbox="402 394 1214 457">• Menghapus dukungan untuk Windows Server 2008 R2.<li data-bbox="402 478 1214 646">• Menetapkan Low Latency Queuing (LLQ) agar selalu aktif untuk meningkatkan performa pada instans Amazon EC2 generasi keenam. <p data-bbox="402 751 613 785">Perbaikan Bug</p> <ul data-bbox="402 835 1214 1318" style="list-style-type: none"><li data-bbox="402 835 1214 1003">• Memperbaiki kegagalan untuk mempublikasikan metrik performa jaringan ke sistem Penghitung Performa untuk Windows (PCW).<li data-bbox="402 1024 1214 1129">• Memperbaiki kebocoran memori selama operasi pembacaan kunci registri.<li data-bbox="402 1150 1214 1318">• 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	<p data-bbox="402 254 610 289">Pengumuman</p> <p data-bbox="402 333 1192 512">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:</p> <ul data-bbox="402 562 1203 810" style="list-style-type: none"><li data-bbox="402 562 766 625">• Instal versi sebelumnya<li data-bbox="435 667 1203 751">1. Unduh paket versi sebelumnya dari tautan di tabel ini (versi 2.2.3).<li data-bbox="435 772 1154 810">2. Jalankan skrip install.ps1 PowerShell instalasi. <p data-bbox="435 919 1133 1050">Untuk detail lebih lanjut untuk langkah-langkah sebelum dan sesudah instalasi lihat Mengaktifkan jaringan yang ditingkatkan di Windows.</p> <p data-bbox="435 1094 1187 1171">Menggunakan Amazon EC2 Systems Manager untuk pembaruan massal</p> <ul data-bbox="435 1220 1179 1465" style="list-style-type: none"><li data-bbox="435 1220 1179 1350">• Lakukan pembaruan massal melalui dokumen <code>AWS-ConfigureAWSPackage</code> SSM, dengan parameter berikut:<ul data-bbox="496 1371 951 1465" style="list-style-type: none"><li data-bbox="496 1371 951 1409">• Nama: <code>AwsEnaNetworkDriver</code><li data-bbox="496 1430 691 1465">• Versi: 2.2.3	26 Oktober 2021

Versi driver	Detail	Tanggal rilis
2.2.3	<p>Fitur baru</p> <ul style="list-style-type: none">• Menambahkan dukungan untuk kartu Nitro baru dengan jaringan instans 400 Gbps. <p>Perbaikan Bug</p> <ul style="list-style-type: none">• Memperbaiki kondisi balapan antara perubahan waktu sistem dan kueri waktu sistem oleh driver ENA, yang menyebabkan deteksi positif palsu dari tidak responsifnya HW. <p>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.</p>	25 Maret 2021

Versi driver	Detail	Tanggal rilis
2.2.2	<p>Fitur Baru</p> <ul style="list-style-type: none">• Menambahkan dukungan ke metrik kinerja adaptor jaringan kueri dengan CloudWatch dan Penghitung Kinerja untuk konsumen Windows. <p>Perbaikan Bug</p> <ul style="list-style-type: none">• Memperbaiki masalah performa pada instans bare metal.	21 Desember 2020
2.2.1	<p>Fitur baru</p> <ul style="list-style-type: none">• Tambahkan metode agar host bisa mengueri Elastic Network Adapter untuk metrik performa jaringan.	1 Oktober 2020

Versi driver	Detail	Tanggal rilis
2.2.0	<p>Fitur Baru</p> <ul style="list-style-type: none">• Menambahkan dukungan untuk tipe perangkat keras generasi selanjutnya.• Meningkatkan waktu mulai instans setelah melanjutkan dari stop-hibernate, dan menghapus pesan kesalahan ENA positif palsu. <p>Optimalisasi Performa</p> <ul style="list-style-type: none">• Mengoptimalkan pemrosesan lalu lintas masuk.• Meningkatkan manajemen memori bersama di lingkungan dengan sumber daya rendah. <p>Perbaikan Bug</p> <ul style="list-style-type: none">• Menghindari sistem crash saat penghapusan perangkat ENA dalam skenario yang jarang terjadi di mana driver gagal untuk mereset.	12 Agustus 2020
2.1.5	<p>Perbaikan Bug</p> <ul style="list-style-type: none">• Memperbaiki kegagalan inisialisasi adaptor jaringan yang sesekali terjadi pada instans bare metal.	23 Juni 2020

Versi driver	Detail	Tanggal rilis
2.1.4	<p>Perbaikan Bug</p> <ul style="list-style-type: none">• Mencegah masalah konektivitas yang disebabkan oleh metadata paket LSO yang rusak dari tumpukan jaringan.• Mencegah kerusakan sistem yang disebabkan oleh kondisi race langka yang mengakibatkan pengaksesan memori paket yang sudah dirilis.	25 November 2019
2.1.2	<p>Fitur baru</p> <ul style="list-style-type: none">• Menambahkan dukungan untuk laporan ID vendor agar OS bisa menghasilkan UUID berbasis MAC. <p>Perbaikan Bug</p> <ul style="list-style-type: none">• Peningkatan performa konfigurasi jaringan DHCP selama inisialisasi.• Menghitung dengan benar checksum L4 pada lalu lintas masuk IPv6 ketika unit transmisi maksimum (MTU) melebihi 4K.• Peningkatan umum untuk stabilitas driver dan perbaikan bug kecil.	4 November 2019

Versi driver	Detail	Tanggal rilis
2.1.1	<p>Perbaikan Bug</p> <ul style="list-style-type: none">• Mencegah penurunan paket TCP LSO yang sangat terfragmentasi, yang datang dari sistem operasi.• 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	<p>ENA Windows driver v2.1 memperkenalkan kapabilitas perangkat ENA baru, memberikan peningkatan performa, menambahkan fitur baru, dan memasukkan beberapa peningkatan stabilitas.</p> <ul style="list-style-type: none">• Fitur baru<ul style="list-style-type: none">• Menggunakan kunci registri Windows standar untuk konfigurasi frame Jumbo.• Mengizinkan pengaturan VLAN ID melalui GUI properti driver ENA.• Alur Pemulihan yang Ditingkatkan<ul style="list-style-type: none">• Mekanisme identifikasi kegagalan yang ditingkatkan.• Menambahkan dukungan untuk parameter pemulihan yang bisa disesuaikan.• Support hingga 32 antrean I/O untuk instans EC2 yang lebih baru yang memiliki lebih dari 8 vCPU.• ~90% pengurangan jejak memori driver.• Optimalisasi performa<ul style="list-style-type: none">• Penurunan latensi jalur transmisi.• Dukungan untuk menerima checksum offload.• Pengoptimalan performa untuk sistem dengan beban berat (penggunaan mekanisme penguncian yang dioptimalkan).	1 Juli 2019

Versi driver	Detail	Tanggal rilis
	<ul style="list-style-type: none">• Penyempurnaan lebih lanjut untuk mengurangi pemakaian CPU dan meningkatkan responsivitas sistem saat diberi beban.• Perbaiki Bug<ul style="list-style-type: none">• Memperbaiki crash karena penguraian yang tidak valid dari header Tx yang tidak berdekatan.• Memperbaiki crash driver v1.5 selama pelepasan antarmuka jaringan elastis pada instans Bare Metal.• Memperbaiki error perhitungan checksum pseudo-header LSO di atas IPv6.• Memperbaiki potensi kebocoran sumber daya memori setelah kegagalan inisialisasi.• Menonaktifkan offload checksum TCP/UDP untuk fragmen IPv4.• Memperbaiki konfigurasi VLAN. VLAN salah dinonaktifkan ketika hanya prioritas VLAN yang seharusnya dinonaktifkan.• Mengaktifkan penguraian yang benar dari pesan driver kustom oleh pelihat peristiwa.• Memperbaiki kegagalan untuk menginisialisasi driver karena penanganan stempel waktu yang tidak valid.• Memperbaiki kondisi race antara pemrosesan data dan penonaktifan perangkat ENA.	

Versi driver	Detail	Tanggal rilis
1.5.0	<ul style="list-style-type: none"> • Peningkatan stabilitas dan perbaikan performa. • Buffer Terima sekarang dapat dikonfigurasi hingga nilai 8192 di Properti Lanjutan ENA NIC. • 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

Amazon SNS dapat memberi Anda notifikasi saat EC2 Windows Drivers versi baru dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Untuk berlangganan pemberitahuan 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-east-1:801119661308:ec2-windows-drivers
```
 - b. Untuk Protokol, pilih Email.

- c. Untuk Titik Akhir, masukkan alamat email yang dapat Anda gunakan untuk menerima pemberitahuan.
 - 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 Langganan.
3. Pilih kotak centang untuk berlangganan lalu pilih Tindakan, Hapus berlangganan. Ketika diminta untuk mengonfirmasi, pilih Hapus.

Tingkatkan kinerja jaringan dengan ENA Express pada instans EC2 Anda

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 Availability Zone yang sama.

Manfaat ENA Ekspres

- Meningkatkan bandwidth maksimum yang dapat digunakan aliran tunggal dari 5 Gbps menjadi 25 Gbps dalam subnet, hingga batas instans agregat.
- Mengurangi latensi ekor lalu lintas jaringan antara instans EC2, terutama selama periode beban jaringan yang 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 mengirim atau menerima volume paket yang tinggi per detik, dan perlu mengoptimalkan latensi sebagian besar waktu, terutama selama periode ketika tidak ada kemacetan di jaringan, [Jaringan yang ditingkatkan](#) mungkin lebih cocok untuk jaringan Anda.

Selama periode waktu ketika lalu lintas jaringan ringan, Anda mungkin melihat sedikit peningkatan latensi paket (puluhan mikrodetik) ketika paket menggunakan ENA Ekspres. 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 menjadi 25 Gbps dalam Availability Zone 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 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.

Prasyarat untuk instance Linux

Untuk memastikan bahwa ENA Express dapat beroperasi secara efektif, perbarui pengaturan untuk instans Anda sebagai berikut.

- Jika instans Anda menggunakan bingkai jumbo, jalankan perintah berikut untuk menyetel unit transmisi maksimum (MTU) ke 8900

```
[ec2-user ~]$ sudo ip link set dev eth0 mtu 8900
```

- Tingkatkan ukuran cincin penerima (Rx), sebagai berikut:

```
[ec2-user ~]$ ethtool -G device rx 8192
```

- Untuk memaksimalkan bandwidth ENA Ekspres, konfigurasi batas antrean TCP Anda sebagai berikut:

1. Tetapkan batas antrian kecil TCP ke 1MB atau lebih tinggi. Ini meningkatkan jumlah data yang antri untuk transmisi pada soket.

```
sudo sh -c 'echo 1048576 > /proc/sys/net/ipv4/tcp_limit_output_bytes'
```

2. Nonaktifkan batas antrian byte pada perangkat eth jika diaktifkan untuk distribusi Linux Anda. Ini meningkatkan antrian data untuk transmisi untuk antrian perangkat.

```
sudo sh -c 'for txq in /sys/class/net/eth0/queues/tx-*; do echo max > ${txq}/byte_queue_limits/limit_min; done'
```

Note

Driver ENA untuk distribusi Amazon Linux menonaktifkan batas antrian byte secara default.

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.

- Instance pengiriman dan penerimaan harus berjalan di Availability Zone 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 instance Linux) Untuk memanfaatkan potensi bandwidth penuh, gunakan driver versi 2.2.9 atau lebih tinggi.
- (Hanya instance Linux) Untuk menghasilkan metrik, gunakan driver versi 2.8 atau lebih tinggi.

Jika ada persyaratan yang tidak terpenuhi, instans menggunakan protokol TCP/UDP 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 dan Pengoptimalan Kinerja Driver ENA Linux](#) 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.

Tipe instans yang didukung untuk ENA Ekspres

Tab berikut menunjukkan jenis instance yang mendukung ENA Express.

General purpose

Jenis 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

Jenis instans	Arsitektur
m6id.metal	x86_64
m7g.12xlarge	arm64
m7g.16xlarge	arm64
m7g.metal	arm64
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-24x1	x86_64
m7i.metal-48x1	x86_64

Compute optimized

Jenis instans	Arsitektur
c6a.12xlarge	x86_64
c6a.16xlarge	x86_64
c6a.24xlarge	x86_64
c6a.32xlarge	x86_64

Jenis instans	Arsitektur
c6a.48xlarge	x86_64
c6a.metal	x86_64
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
c7g.12xlarge	arm64
c7g.16xlarge	arm64
c7g.metal	arm64
c7gd.12xlarge	arm64
c7gd.16xlarge	arm64

Jenis instans	Arsitektur
c7gd.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

Memory optimized

Jenis 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
r6i.8xlarge	x86_64
r6i.12xlarge	x86_64
r6i.16xlarge	x86_64
r6i.24xlarge	x86_64

Jenis instans	Arsitektur
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
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
r7i.metal-24xl	x86_64
r7i.metal-48xl	x86_64

Jenis instans	Arsitektur
u7i-12tb.224xlarge	x86_64
u7in-16tb.224xlarge	x86_64
u7in-24tb.224xlarge	x86_64
u7in-32tb.224xlarge	x86_64
x2idn.16xlarge	x86_64
x2idn.24xlarge	x86_64
x2idn.32xlarge	x86_64
x2idn.metal	x86_64
x2iedn.8xlarge	x86_64
x2iedn.16xlarge	x86_64
x2iedn.24xlarge	x86_64
x2iedn.32xlarge	x86_64
x2iedn.metal	x86_64

Accelerated computing

Jenis instans	Arsitektur
g6.48xlarge	x86_64

Storage optimized

Jenis instans	Arsitektur
i4g.4xlarge	arm64

Jenis instans	Arsitektur
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
im4gn.4xlarge	arm64
im4gn.8xlarge	arm64
im4gn.16xlarge	arm64

Daftar dan lihat pengaturan ENA Ekspres

Bagian ini mencakup cara membuat daftar dan melihat informasi ENA Ekspres dari AWS Management Console atau dari AWS CLI. Untuk informasi lebih lanjut, pilih tab yang cocok dengan metode yang akan Anda gunakan.

Console

Tab ini mencakup cara menemukan informasi tentang pengaturan ENA Ekspres Anda saat ini dan untuk melihat dukungan tipe instans di AWS Management Console.

Lihat dukungan tipe instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Tipe instans.

3. Pilih tipe instans untuk melihat detail untuk instans itu. Anda dapat memilih tautan Tipe 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 tab Jaringan atau bagian itu di halaman detail, dukungan ENA Ekspres menunjukkan nilai benar atau salah untuk menunjukkan apakah tipe instans mendukung fitur ini.

Lihat pengaturan dari daftar 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 Ekspres dan UDP ENA Ekspres.

Lihat pengaturan dari instans

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

Tab ini mencakup cara menemukan informasi tentang pengaturan ENA Ekspres Anda saat ini dan untuk melihat dukungan tipe instans di AWS CLI.

Jelaskan tipe instans

Untuk informasi tentang pengaturan tipe instance untuk tipe instans tertentu, jalankan [describe-instance-types](#) perintah di AWS CLI, dan ganti jenis instance sebagai berikut:

```
[ec2-user ~]$ aws ec2 describe-instance-types --instance-types m6i.metal
{
  "InstanceTypes": [
    {
      "InstanceType": "m6i.metal",
      "CurrentGeneration": true,
      ...
    },
    "NetworkInfo": {
      ...
      "EnaSrdSupported": true
    },
    ...
  ]
}
```

Menjelaskan instans

Untuk informasi tentang konfigurasi ENA Express untuk instance tertentu, jalankan [describe-instances](#) perintah di AWS CLI, sebagai berikut. Contoh perintah ini mengembalikan daftar konfigurasi ENA Express untuk antarmuka jaringan yang dilampirkan ke masing-masing instance yang berjalan yang ditentukan oleh parameter. `--instance-ids`

```
[ec2-user ~]$ aws ec2 describe-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7 --query 'Reservations[*].Instances[*].[InstanceId, NetworkInterfaces[*].Attachment.EnaSrdSpecification]'
```

```
[
  [
    "i-1234567890abcdef0",
    [
      {
        "EnaSrdEnabled": true,
        "EnaSrdUdpSpecification": {
          "EnaSrdUdpEnabled": false
        }
      }
    ]
  ],
  [
    [

```

```

    "i-0598c7d356eba48d7",
    [
      {
        "EnaSrdEnabled": true,
        "EnaSrdUdpSpecification": {
          "EnaSrdUdpEnabled": false
        }
      }
    ]
  ]
]
]
]
]

```

Jelaskan antarmuka jaringan

Untuk informasi tentang pengaturan ENA Express untuk antarmuka jaringan, jalankan [describe-network-interfaces](#) perintah AWS CLI sebagai berikut:

```

[ec2-user ~]$ aws ec2 describe-network-interfaces
{
  "NetworkInterfaces": [
    {
      "Association": {
        ....IPs, DNS...
      },
      "Attachment": {
        "AttachTime": "2022-11-17T09:04:28+00:00",
        "AttachmentId": "eni-attach-0ab1c23456d78e9f0",
        "DeleteOnTermination": true,
        "DeviceIndex": 0,
        "NetworkCardIndex": 0,
        "InstanceId": "i-1234567890abcdef0",
        "InstanceOwnerId": "111122223333",
        "Status": "attached",
        "EnaSrdSpecification": {
          "EnaSrdEnabled": true,
          "EnaSrdUdpSpecification": {
            "EnaSrdUdpEnabled": true
          }
        }
      }
    },
    ...
  ]
}

```

```
"OwnerId": "111122223333",
...
}
]
}
```

PowerShell

Tab ini mencakup cara menemukan informasi tentang pengaturan ENA Express Anda saat ini dan untuk melihat dukungan tipe instans menggunakan PowerShell.

Jelaskan tipe instans

Untuk informasi tentang setelan tipe instans untuk jenis instans tertentu, jalankan [Get-EC2InstanceType Cmdlet](#) dengan Tools for PowerShell, dan ganti jenis instance sebagai berikut:

```
PS C:\> Get-EC2InstanceType -InstanceType m6i.metal | `
Select-Object `
    InstanceType,
    CurrentGeneration,
    @{Name = 'EnaSrdSupported'; Expression = { $_.NetworkInfo.EnaSrdSupported } } | `
Format-List

InstanceType      : m6i.metal
CurrentGeneration : True
EnaSrdSupported   : True
```

Jika ENA Ekspres diaktifkan, nilai `True` dikembalikan.

Jelaskan antarmuka jaringan

Untuk informasi tentang pengaturan ENA Express untuk antarmuka jaringan, jalankan [Get-EC2NetworkInterface Cmdlet](#) dengan Alat untuk PowerShell sebagai berikut:

```
PS C:\> Get-EC2NetworkInterface -NetworkInterfaceId eni-0d1234e5f6a78901b | `
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 } }

```

```

Association          :
NetworkInterfaceId  : eni-0d1234e5f6a78901b
OwnerId             : 111122223333
AttachTime          : 6/11/2022 1:13:11 AM
AttachmentId        : eni-attach-0d1234e5f6a78901b
DeleteOnTermination : True
NetworkCardIndex    : 0
InstanceId           : i-0d1234e5f6a78901b
InstanceOwnerId     : 111122223333
Status              : attached
EnaSrdEnabled       : True
EnaSrdUdpEnabled    : False

```

Konfigurasi pengaturan ENA Ekspres

Anda dapat mengonfigurasi ENA Ekspres untuk tipe instans EC2 yang didukung tanpa perlu menginstal perangkat lunak tambahan apa pun.

Bagian ini mencakup cara mengkonfigurasi ENA Express dari AWS Management Console atau dari AWS CLI. Untuk informasi lebih lanjut, pilih tab yang cocok dengan metode yang akan Anda gunakan.

Console

Tab ini mencakup cara mengelola pengaturan ENA Ekspres untuk antarmuka jaringan yang dilampirkan ke sebuah instans.

Kelola ENA Ekspres dari daftar 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.

 Note

Jika antarmuka jaringan yang Anda pilih tidak dilampirkan ke sebuah instans, tindakan ini tidak muncul di menu.

5. Untuk menggunakan ENA Ekspres, pilih kotak centang Aktifkan.
6. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Ekspres UDP, pilih kotak centang Aktifkan.
7. Untuk menyimpan pengaturan Anda, pilih Simpan.

Kelola ENA Ekspres dari daftar Instans

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 Ekspres untuk lampiran antarmuka jaringan yang dipilih, pilih kotak centang Aktifkan.
8. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Ekspres UDP, pilih kotak centang Aktifkan.
9. Untuk menyimpan pengaturan Anda, pilih Simpan.

Mengonfigurasi ENA Ekspres saat Anda memasang antarmuka jaringan ke instans EC2

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 Ekspres setelah Anda melampirkan antarmuka jaringan ke instans, pilih kotak centang Aktifkan.
6. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Ekspres UDP, pilih kotak centang Aktifkan.
7. Untuk melampirkan antarmuka jaringan ke instans dan menyimpan pengaturan ENA Ekspres Anda, pilih Lampirkan.

AWS CLI

Tab ini mencakup cara mengonfigurasi pengaturan ENA Ekspres di AWS CLI.

Konfigurasi ENA Ekspres saat Anda memasang antarmuka jaringan

Untuk mengkonfigurasi ENA Express saat Anda melampirkan antarmuka jaringan ke sebuah instance, jalankan [attach-network-interface](#) perintah di AWS CLI, seperti yang ditunjukkan pada contoh berikut:

Contoh 1: Gunakan ENA Ekspres untuk lalu lintas TCP, tetapi tidak untuk lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` sebagai `true`, dan kami mengizinkan default `EnaSrdUdpEnabled` ke `false`.

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-srd-specification 'EnaSrdEnabled=true'
{
  "AttachmentId": "eni-attach-012c3d45e678f9012"
}
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai `true`.

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-srd-specification 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'  
{  
  "AttachmentId": "eni-attach-012c3d45e678f9012"  
}
```

Perbarui pengaturan ENA Ekspres untuk lampiran antarmuka jaringan Anda

Untuk memperbarui pengaturan ENA Express untuk antarmuka jaringan yang dilampirkan ke sebuah instance, jalankan [modify-network-interface-attribute](#) perintah di AWS CLI, seperti yang ditunjukkan pada contoh berikut:

Contoh 1: Gunakan ENA Ekspres untuk lalu lintas TCP, tetapi tidak untuk lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` sebagai `true`, dan kami mengizinkan default `EnaSrdUdpEnabled` ke `false` jika belum pernah disetel sebelumnya.

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --network-interface-id eni-0123f4567890a1b23 --ena-srd-specification 'EnaSrdEnabled=true'
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai `true`.

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --network-interface-id eni-0123f4567890a1b23 --ena-srd-specification 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
```

Contoh 3: Berhenti menggunakan ENA Ekspres untuk lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdUdpEnabled` sebagai `false`.

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --network-interface-id eni-0123f4567890a1b23 --ena-srd-specification 'EnaSrdUdpSpecification={EnaSrdUdpEnabled=false}'
```

PowerShell

Tab ini mencakup cara mengkonfigurasi pengaturan ENA Express menggunakan PowerShell.

Konfigurasi ENA Ekspres saat Anda memasang antarmuka jaringan

Untuk mengkonfigurasi pengaturan ENA Express untuk antarmuka jaringan, jalankan [Add-EC2NetworkInterface Cmdlet](#) dengan Alat untuk PowerShell seperti yang ditunjukkan pada contoh berikut:

Contoh 1: Gunakan ENA Ekspres untuk lalu lintas TCP, tetapi tidak untuk lalu lintas UDP

Dalam contoh ini, kami mengkonfigurasi `EnaSrdEnabled` sebagai `true`, dan kami mengizinkan default `EnaSrdUdpEnabled` ke `false`.

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true

eni-attach-012c3d45e678f9012
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Dalam contoh ini, kami mengkonfigurasi `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai `true`.

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdUdpSpecification_EnaSrdUdpEnabled $true

eni-attach-012c3d45e678f9012
```

Perbarui pengaturan ENA Ekspres untuk lampiran antarmuka jaringan Anda

Untuk memperbarui pengaturan ENA Express untuk antarmuka jaringan yang dilampirkan ke instance, jalankan [Add-EC2NetworkInterface Cmdlet](#) perintah di Alat untuk PowerShell, seperti yang ditunjukkan dalam contoh berikut:

Contoh 1: Gunakan ENA Ekspres untuk lalu lintas TCP, tetapi tidak untuk lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` sebagai `true`, dan kami mengizinkan default `EnaSrdUdpEnabled` ke `false` jika belum pernah disetel sebelumnya.

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-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

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : False
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai `true`.

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
    { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
    { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : True
```

Contoh 3: Berhenti menggunakan ENA Ekspres untuk lalu lintas UDP

Dalam contoh ini, kami mengonfigurasi `EnaSrdUdpEnabled` sebagai `false`.

```
PS C:\> 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

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : False
```

Konfigurasi ENA Express saat peluncuran

Anda dapat menggunakan salah satu metode berikut untuk mengonfigurasi ENA Ekspres untuk AMI saat Anda meluncurkan sebuah instans dari AWS Management Console.

- Anda dapat mengonfigurasi ENA Ekspres untuk AMI saat meluncurkan instans dengan wizard peluncuran instans. Untuk detail konfigurasi, lihat Konfigurasi jaringan lanjutan di [Pengaturan jaringan](#) untuk wizard peluncuran instans.
- Anda dapat mengonfigurasi ENA Ekspres untuk AMI Anda saat Anda menggunakan template peluncuran. Untuk informasi selengkapnya tentang konfigurasi template peluncuran, lihat Konfigurasi jaringan lanjutan di templat [Pengaturan jaringan](#) untuk peluncuran.

Pantau performa ENA Ekspres

Setelah mengaktifkan ENA Ekspres untuk lampiran antarmuka jaringan pada instans pengiriman dan instans penerima, Anda dapat menggunakan metrik ENA Ekspres untuk membantu memastikan bahwa instans Anda memanfaatkan sepenuhnya peningkatan performa yang disediakan oleh teknologi SRD.

Untuk melihat daftar metrik yang difilter untuk ENA Ekspres, jalankan perintah `ethtool` berikut untuk antarmuka jaringan Anda (ditampilkan di sini sebagai `eth0`):

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
  ena_srd_mode: 0
  ena_srd_tx_pkts: 0
  ena_srd_eligible_tx_pkts: 0
  ena_srd_rx_pkts: 0
  ena_srd_resource_utilization: 0
```

Verifikasi pengaturan ENA Ekspres untuk sebuah instans

Untuk memverifikasi pengaturan ENA Ekspres saat ini untuk lampiran antarmuka jaringan pada instans Anda, jalankan perintah `ethtool` untuk mencantumkan metrik ENA Ekspres, dan perhatikan nilai `ena_srd_mode` metrik. 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

Setelah Anda mengaktifkan ENA Ekspres untuk lampiran antarmuka jaringan pada sebuah instans, instans pengirim memulai komunikasi dengan instans penerima, dan SRD mendeteksi apakah ENA Ekspres 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. 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.

Anda dapat memantau utilisasi sumber daya SRD Anda menggunakan metrik.

`ena_srd_resource_utilization` Jika instans Anda hampir menghabiskan sumber daya SRD-nya, Anda akan tahu sudah waktunya untuk menskalakan ke luar instans.

Untuk informasi lebih lanjut tentang metrik ENA Ekspres, lihat [Metrik untuk ENA Ekspres](#).

Tune kinerja untuk pengaturan ENA Express

Untuk memeriksa konfigurasi instans Linux Anda untuk kinerja ENA Express yang optimal, Anda dapat menjalankan skrip berikut yang tersedia di GitHub repositori Amazon:

<https://github.com/amzn/amzn-ec2-ena-utilities/blob/main/ena-express/check-ena-express-settings.sh>

Skrip menjalankan serangkaian tes dan menyarankan perubahan konfigurasi yang direkomendasikan dan yang diperlukan.

Aktifkan jaringan yang disempurnakan dengan antarmuka Intel 82599 VF pada instans EC2

Amazon EC2 memberikan kemampuan jaringan yang ditingkatkan melalui antarmuka Intel 82599 VF, yang menggunakan driver `ixgbevf` Intel.

Daftar Isi

- [Persyaratan](#)
- [Verifikasi bahwa driver telah diinstal](#)
- [Menguji apakah jaringan yang ditingkatkan diaktifkan](#)
- [Mengaktifkan jaringan yang ditingkatkan pada instans Anda](#)
- [Memecahkan masalah konektivitas](#)

Persyaratan

Untuk mempersiapkan jaringan yang ditingkatkan menggunakan antarmuka Intel 82599 VF, siapkan instans Anda sebagai berikut:

- Pilih dari tipe instans yang didukung berikut: 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.

- Instance 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 di 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 menyarankan Anda memperbarui driver tersebut ke versi terbaru menggunakan utilitas Pnputil.exe.
- Gunakan [AWS CloudShell](#) dari AWS Management Console, atau instal dan konfigurasi [AWS CLI](#) atau [AWS Tools for Windows PowerShell](#) di 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.

Verifikasi bahwa driver telah diinstal

Verifikasi bahwa driver diinstal pada instance Anda.

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 `ixgbev` 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
```

Adaptor jaringan Windows

Untuk memverifikasi bahwa driver telah diinstal, sambungkan ke instans Anda dan buka Pengelola Perangkat. Anda seharusnya melihat "Fungsi Virtual Intel (R) 82599" tercantum di Adaptor jaringan.

Menguji apakah jaringan yang ditingkatkan diaktifkan

Verifikasi bahwa `sriovNetSupport` atribut diatur.

Atribut contoh (`sriovNetSupport`)

Untuk memeriksa apakah sebuah instans memiliki set atribut `sriovNetSupport` jaringan yang ditingkatkan, gunakan salah satu dari perintah berikut. Jika atribut diatur, nilainya adalah `simple`.

- [describe-instance-attribute](#) (AWS CLI) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instance-attribute --instance-id instance_id --attribute sriovNetSupport
```

- [Get-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Get-EC2InstanceAttribute -InstanceId instance-id -Attribute sriovNetSupport
```

Atribut gambar (sriovNetSupport)

Untuk memeriksa apakah AMI sudah memiliki set `sriovNetSupport` atribut jaringan yang disempurnakan, gunakan salah satu perintah berikut. Jika atribut diatur, nilainya adalah `simple`.

- [describe-images](#) (AWS CLI)

```
aws ec2 describe-images --image-id ami_id --query "Images[].SriovNetSupport"
```

- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

```
(Get-EC2Image -ImageId ami-id).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, reboot instans Anda menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [reboot-instances](#) (AWS CLI), [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. [Instans yang didukung EBS] Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu dari perintah berikut: [stop-instances](#) (AWS CLI), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

[Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebagai gantinya, lanjutkan ke prosedur ini: [Untuk mengaktifkan jaringan yang ditingkatkan \(instans yang didukung penyimpanan instans\)](#).

6. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

AWS CLI

[modify-instance-attribute](#) (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

PowerShell

[Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

7. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
9. Hubungkan ke instans Anda dan verifikasi bahwa modul `ixgbevf` diinstal dan dimuat di antarmuka jaringan Anda menggunakan perintah `ethtool -i ethn` 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 AMI Linux yang didukung penyimpanan instans](#), pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

AWS CLI

[register-image](#) (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

[Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
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 **linux-aws** paket kernel

1. Connect 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. Connect ke instans Anda.

2. Unduh sumber untuk modul `ixgbevf` di instans 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 versi `systemd` atau `udev` pada sistem berbasis RPM 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. [Instans yang didukung EBS] Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu dari perintah berikut: [stop-instances](#) (AWS CLI/AWS CloudShell), [Stop-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

[Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebagai gantinya, lanjutkan ke prosedur ini: [Untuk mengaktifkan jaringan yang ditingkatkan \(instans yang didukung penyimpanan instans\)](#).

8. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

AWS CLI

[modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

PowerShell

[Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

9. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
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 AMI Linux yang didukung penyimpanan instans](#), pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

AWS CLI

[register-image](#) (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

[Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
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 [Dukungan jaringan yang ditingkatkan](#).

⚠ Important

Untuk melihat pembaruan driver terbaru di AMI Windows, lihat [riwayat versi Windows AMI](#) di Referensi AMI AWS 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. Dalam folder yang diekstrak, cari file `Wired_driver_version_x64.exe` dan ganti namanya menjadi `Wired_driver_version_x64.zip`.
3. Ekstrak isi file `Wired_driver_version_x64.zip` tersebut.
4. 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

```
C:\> pnputil -i -a PROXGB\Winx64\WS2022\vxS.inf
```

- Windows Server 2019

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
```

- Windows Server 2016

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS65\vxn65x64.inf
```

- Windows Server 2012 R2

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS64\vxn64x64.inf
```

- Windows Server 2012

```
C:\> 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

[modify-instance-attribute](#) (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support simple
```

PowerShell

[Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

6. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di [Buat AMI yang didukung Amazon EBS-Backed](#). 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 instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (AWS CLI), [Start-EC2Instance](#) (AWS Tools for Windows PowerShell). Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

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 performa jaringan untuk 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 hal-hal 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 instans menggunakan mekanisme kredit I/O jaringan untuk mengalokasikan bandwidth jaringan berdasarkan penggunaan bandwidth rata-rata. Amazon EC2 juga memiliki bandwidth maksimum untuk lalu lintas ke AWS Direct Connect dan internet. Untuk informasi selengkapnya, lihat [Bandwidth jaringan instans Amazon EC2](#).
- Kinerja Packet-per-second (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](#)
- Akses layanan link-local — Amazon EC2 menyediakan PPS maksimum per antarmuka jaringan untuk lalu lintas ke layanan seperti layanan DNS, Layanan Metadata Instans, dan Layanan Amazon Time Sync.

Ketika lalu lintas jaringan untuk suatu instance melebihi maksimum, AWS membentuk 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 menjalankan 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 `contrack_allowance_available` metrik, instal driver ENA versi 2.8.1.

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
<code>bw_in_allowance_exceeded</code>	Jumlah paket antri atau dijatuhkan karena kumpulan bandwidth yang masuk melebihi maksimum untuk instans.	Semua tipe instans
<code>bw_out_allowance_exceeded</code>	Jumlah paket antri atau dijatuhkan karena bandwidth agregat yang keluar melebihi maksimum untuk instans.	Semua tipe instans
<code>contrack_allowance_exceeded</code>	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.	Semua tipe instans
<code>contrack_allowance_available</code>	Jumlah koneksi yang dilacak yang dapat dibuat oleh instans	Instans yang dibangun di atas Sistem AWS Nitro saja

Metrik	Deskripsi	Didukung pada
	sebelum menekan tunjangan Connections Tracked dari tipe instans tersebut.	
<code>linklocal_allowance_exceeded</code>	Jumlah paket turun karena PPS lalu lintas ke layanan proksi lokal melebihi batas maksimum untuk antarmuka jaringan. Hal ini berdampak lalu lintas ke layanan DNS, Layanan Metadata Instans, dan Layanan Amazon Time Sync.	Semua tipe instans
<code>pps_allowance_exceeded</code>	Jumlah paket yang diantrekan atau dijatuhkan karena PPS dua arah melebihi maksimum untuk instans.	Semua tipe instans

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
```

```
contrack_allowance_available: 136812
```

Instans Windows

Anda dapat melihat metrik menggunakan pengukur performa Windows. Data dapat diuraikan 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. Anda dapat menggunakan metrik ENA Ekspres untuk

membantu memastikan bahwa instans Anda memanfaatkan sepenuhnya peningkatan performa yang disediakan oleh teknologi SRD, misalnya:

- 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.

Metrik ENA Ekspres berikut tersedia menggunakan ethtool perintah untuk instans berbasis Linux.

- `ena_srd_mode` – Menjelaskan fitur ENA Ekspres 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 paket jaringan yang dikirim dalam jangka waktu tertentu yang memenuhi persyaratan kelayakan SRD, 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.
 - Instance pengiriman dan penerimaan harus berjalan di Availability Zone 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 diterima dalam jangka waktu tertentu.
- `ena_srd_rx_pkts` – Jumlah paket SRD yang diterima dalam jangka waktu tertentu.
- `ena_srd_resource_utilization` – Persentase pemanfaatan memori maksimum yang diizinkan untuk koneksi SRD bersamaan yang telah dikonsumsi instans.

Untuk melihat daftar metrik yang difilter untuk ENA Ekspres, jalankan perintah `ethtool` berikut untuk antarmuka jaringan Anda (ditampilkan di sini sebagai `eth0`):

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
  ena_srd_mode: 0
  ena_srd_tx_pkts: 0
  ena_srd_eligible_tx_pkts: 0
  ena_srd_rx_pkts: 0
  ena_srd_resource_utilization: 0
```

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 instans telah menggunakan sumber daya maksimumnya, atau jika paket melebihi batas MTU, paket yang memenuhi syarat tidak dapat ditransmisikan 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 suatu instance melebihi maksimum, AWS membentuk 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.

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
contrack_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 menjalankan FreeBSD

Dimulai dengan versi 2.3.0, driver ENA FreeBSD mendukung pengumpulan metrik performa jaringan pada instans yang menjalankan FreeBSD. Untuk mengaktifkan koleksi metrik FreeBSD, masukkan perintah berikut dan *atur interval ke nilai antara 1 dan 3600*. Ini menentukan seberapa sering, dalam hitungan detik, untuk mengumpulkan metrik FreeBSD.

```
sysctl dev.ena.network_interface.eni_metrics.sample_interval=interval
```

Sebagai contoh, perintah berikut menetapkan driver untuk mengumpulkan metrik FreeBSD pada antarmuka jaringan 1 setiap 10 detik:

```
sysctl dev.ena.1.eni_metrics.sample_interval=10
```

Untuk menonaktifkan koleksi metrik FreeBSD, Anda dapat menjalankan perintah sebelumnya dan menentukan 0 sebagai intervalnya.

Setelah mengaktifkan pengumpulan metrik FreeBSD, Anda dapat mengambil kumpulan metrik terbaru yang dikumpulkan dengan menjalankan perintah berikut.

```
sysctl dev.ena.network_interface.eni_metrics
```

Memecahkan Masalah Adaptor Jaringan Elastis 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 saya memigrasikan instans EC2 saya ke instance 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](#)
- [Mekanisme keep-alive](#)
- [Mendaftarkan waktu habis baca](#)
- [Statistik](#)
- [Log error driver di syslog](#)
- [Pemberitahuan konfigurasi sub-optimal](#)

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 (instans yang didukung EBS)

1. Dari komputer lokal Anda, hentikan instans menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [stop-instances](#) (), ()AWS CLI. [Stop-EC2Instance](#)AWS Tools for Windows PowerShell Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus menghentikan instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.

⚠ Important

Jika Anda menggunakan instans yang didukung penyimpanan instans, Anda tidak dapat menghentikan instans tersebut. Sebaliknya, lanjutkan ke [Untuk menonaktifkan jaringan yang ditingkatkan dengan ENA \(instans yang didukung penyimpanan instans\)](#).

2. Dari komputer lokal Anda, nonaktifkan atribut jaringan yang ditingkatkan menggunakan perintah berikut.

- [modifikasi-atribut-instans](#) (AWS CLI)

```
$ C:\> aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
```

3. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: [start-instances](#) (), ()AWS CLI. [Start-EC2Instance](#)AWS Tools for Windows PowerShell Jika instance Anda dikelola oleh AWS OpsWorks, Anda harus memulai instance di AWS OpsWorks konsol sehingga status instance tetap sinkron.
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 Elastic Network Adapter \(ENA\) pada instans EC2](#).

Untuk menonaktifkan jaringan yang ditingkatkan dengan ENA (instans yang didukung penyimpanan instans)

Jika instans Anda adalah instans yang didukung penyimpanan instans, buat AMI baru seperti yang dijelaskan di [Buat AMI Linux yang didukung penyimpanan instans](#). Pastikan untuk menonaktifkan atribut `enaSupport` jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

- [register-image](#) (AWS CLI)

```
$ C:\> aws ec2 register-image --no-ena-support ...
```

- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

```
C:\> Register-EC2Image -EnaSupport $false ...
```

Mekanisme keep-alive

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 pembacaan I/O (MMIO) yang dipetakan memori. 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
contrack_allowance_available: 450878
contrack_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 antre atau dijatuhkan karena bandwidth agregat yang keluar melebihi maksimum untuk instans.

`pps_allowance_exceeded`: *N*

Jumlah paket yang diantrekan atau dijatuhkan karena PPS dua arah melebihi maksimum untuk instans.

`contrack_allowance_available`: *N*

Jumlah koneksi yang dilacak yang dapat dibuat oleh instans sebelum menekan tunjangan Connections Tracked dari tipe instans tersebut. Hanya tersedia untuk instans berbasis Nitro. Tidak didukung dengan instans FreeBSD atau lingkungan DPDK.

`contrack_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. Hal ini berdampak lalu lintas ke layanan DNS, Layanan Metadata Instans, dan Layanan Amazon Time Sync.

`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`

Jumlah berapa kali antrean *N* penuh dan dihentikan.

`queue_N_tx_queue_wakeup: N`

Jumlah berapa kali antrean *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 melewati 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 RSS Indirection tidak didukung untuk perangkat ini.

- **18:** Konfigurasi RSS Hash Input tidak didukung untuk perangkat ini.
- **20:** Moderasi interupsi tidak didukung untuk perangkat ini.
- **27:** Driver Elastic Network Adapter tidak mendukung polling kapabilitas 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.

Pemberitahuan konfigurasi sub-optimal

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.

- **Kode1:** 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.

```
sudo rmmod ena && sudo modprobe ena force_large_llq_header=0
```

Untuk informasi selengkapnya tentang konfigurasi optimal untuk ENA Ekspres, lihat [Tingkatkan kinerja jaringan dengan ENA Express pada instans EC2 Anda](#).

- **Kode2:** 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.

Anda dapat menjalankan `ethtool` perintah berikut untuk menyesuaikan ukuran LLQ. Untuk mempelajari lebih lanjut tentang cara mengontrol, menanyakan, dan mengaktifkan Wide-LLQ, lihat topik [Large Low-Latency Queue \(Large LLQ\)](#) dari driver kernel Linux untuk dokumentasi 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 [Tingkatkan kinerja jaringan dengan ENA Express pada instans EC2 Anda](#).

- Kode3: 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 menyarankan Anda mengaktifkan Wide-LLQ, yang meningkatkan ukuran header paket Tx yang didukung hingga maksimum 224 byte.

Namun, saat Anda mengaktifkan Wide-LLQ, ukuran cincin Tx maksimum dikurangi dari 1000 menjadi 512 entri. Wide-LLQ diaktifkan secara default untuk semua jenis Nitro v4 dan instance yang lebih baru.

- Jenis instans Nitro v4 memiliki ukuran cincin Wide-LLQ Tx maksimum default 512 entri, yang tidak dapat diubah.
- Jenis instans Nitro v5 memiliki ukuran cincin Wide-LLQ Tx default 512 entri, yang dapat Anda tingkatkan hingga 1000 entri.

Anda dapat menjalankan `ethtool` perintah berikut untuk menyesuaikan ukuran LLQ. Untuk mempelajari lebih lanjut tentang cara mengontrol, menanyakan, dan mengaktifkan Wide-LLQ, lihat topik [Large Low-Latency Queue \(Large LLQ\)](#) dari driver kernel Linux untuk dokumentasi 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 Elastis

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.

Instal driver Elastic Network Adapter (ENA)

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:

- Memiliki PowerShell versi 3.0 atau yang lebih baru diinstal

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 AWS Management Console, 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 instance 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.

- 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

- Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
- Unduh paket driver ENA untuk versi tertentu yang Anda inginkan dari tautan versi di tabel [Pengemudi Windows ENA](#).
- 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.

- 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.
- 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 Elastis](#).

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.

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".

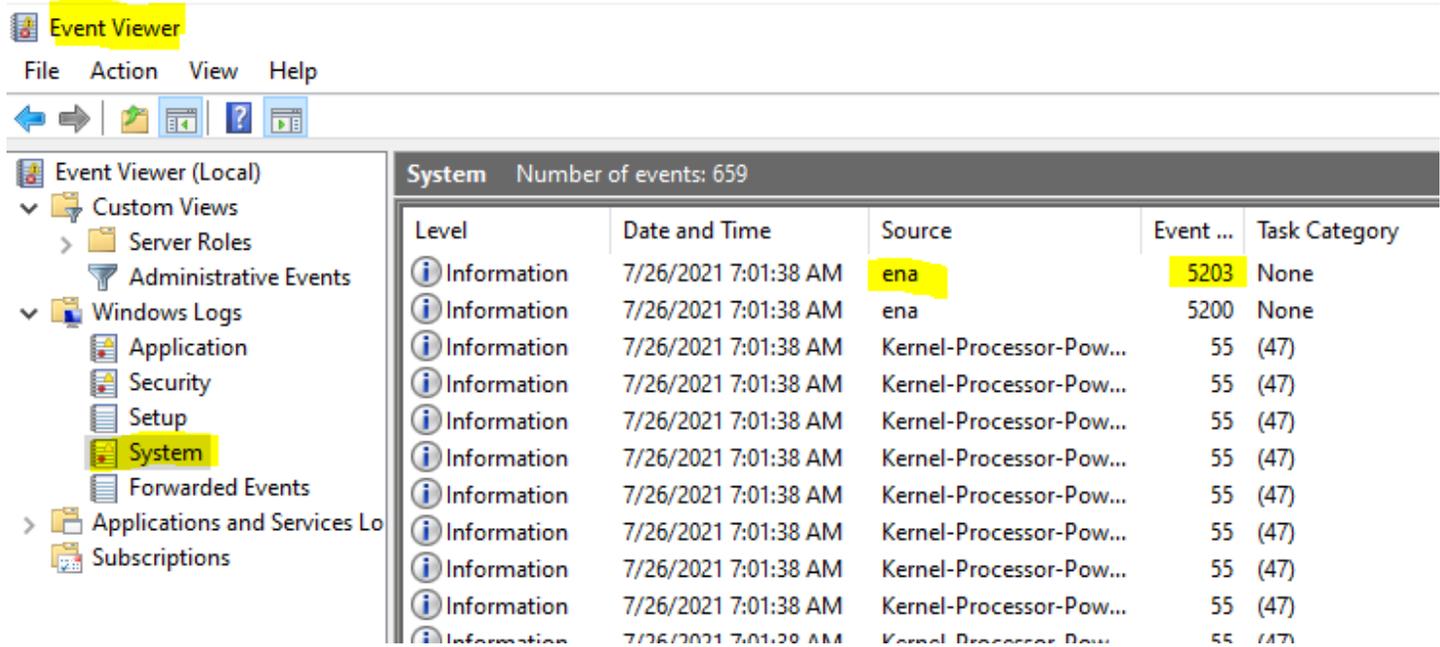
Selidiki pesan peristiwa driver

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

ID peristiwa	Deskripsi peristiwa driver ENA	Tipe
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
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 performa jaringan untuk instans EC2 Anda](#).

Pada contoh di mana metrik ENA diaktifkan, dan CloudWatch agen Amazon diinstal, CloudWatch mengumpulkan metrik yang terkait dengan penghitung di Windows Performance Monitor, 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 CloudWatch](#) Panduan 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.

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

- Pesan yang masih aktif tidak ada

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 I/O (MMIO) yang dipetakan, 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 [Pengemudi Windows ENA](#).

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 Selanjutnya.
5. Jika penginstal tidak secara otomatis me-reboot instance Anda, jalankan Restart-Computer PowerShell cmdlet.

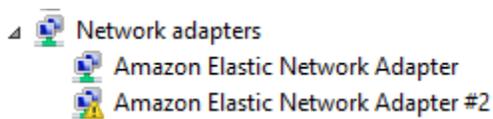
```
PS C:\> Restart-Computer
```

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](#) dalam dokumentasi Developer Perangkat Keras Microsoft Windows.

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. Grafik penggunaan per CPU 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](#).

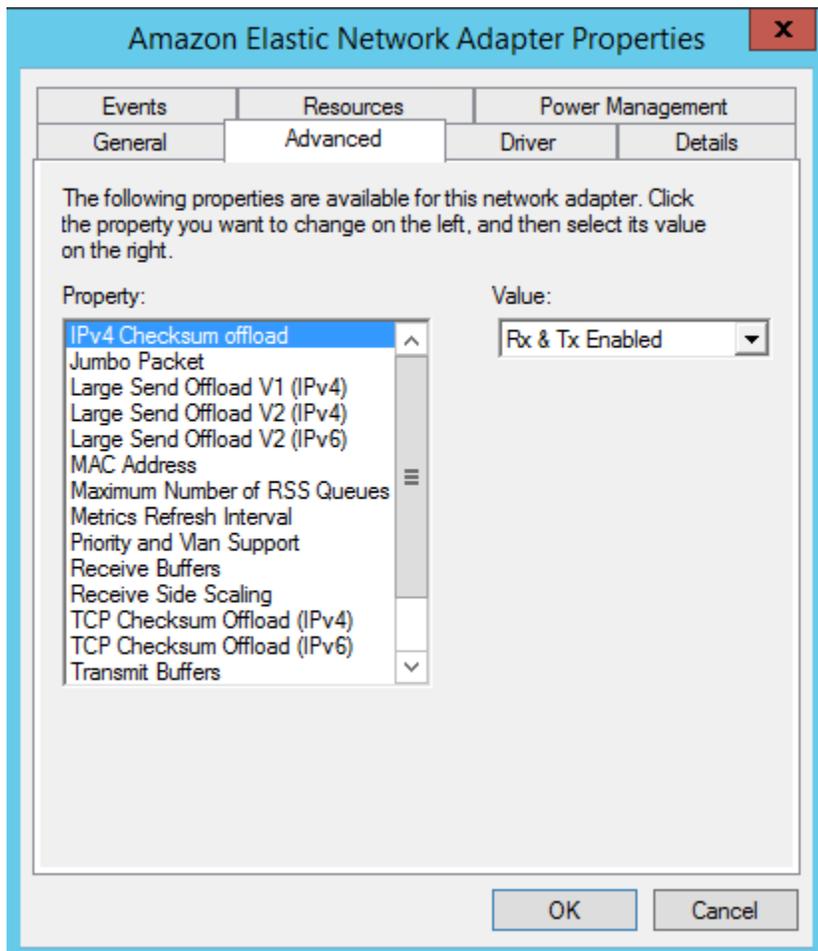
Anda mungkin perlu memodifikasi sistem operasi default untuk mencapai performa jaringan maksimum pada instans dengan jaringan yang ditingkatkan. Beberapa optimasi, seperti mengaktifkan checksum offloading dan mengaktifkan RSS, dikonfigurasi secara default di AMI Windows resmi. Untuk pengoptimalan lain yang dapat Anda terapkan ke adaptor ENA, lihat penyesuaian performa yang ditunjukkan di [Penyesuaian performa adaptor ENA](#).

Kami menyarankan Anda melanjutkan dengan hati-hati, dan membatasi penyesuaian properti perangkat ke yang tercantum di bagian ini, atau perubahan spesifik 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 mempelajari selengkapnya, lihat Optimalkan kinerja jaringan pada instance Windows .

Properti	Deskripsi	Nilai default	Penyesuaian
Jumlah Antrean RSS Maksimum	Mengatur jumlah maksimum antrean RSS yang diizinkan saat RSS diaktifkan.	32	<p>Jumlah antrean RSS ditentukan selama inialisasi pengemudi , dan mencakup batasan berikut (antara lain):</p> <ul style="list-style-type: none">• Batas antrean RSS ditetapkan oleh properti ini• Batas instans (jumlah vCPU)• <p>Batas pembuatan perangkat keras (hingga 8 antrian RSS di eNAV1, dan hingga 32 antrean RSS di eNAV2)</p> <p>Anda dapat mengatur nilai dari 1-32, tergantung pada instans dan batas pembuatan perangkat keras Anda. Untuk mempelajari selengkapnya, lihat Optimalkan kinerja jaringan pada instance Windows.</p>

Properti	Deskripsi	Nilai default	Penyesuaian
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 AWS Wilayah 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:

- Konfigurasi 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 semaksimal mungkin. Untuk informasi selengkapnya, lihat [Grup penempatan](#).
- Anda dapat menggunakan frame jumbo untuk lalu lintas antara VPC dan jaringan on-premise Anda melalui AWS Direct Connect. Untuk informasi selengkapnya tentang penggunaan AWS Direct Connect, dan verifikasi kemampuan jumbo frame, lihat [Mengatur MTU jaringan untuk antarmuka virtual pribadi atau antarmuka virtual transit](#) di Panduan Pengguna AWS Direct Connect
- 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 latensi jaringan untuk instans Amazon 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 Praktik Terbaik dan Pengoptimalan Kinerja Driver ENA Linux](#) GitHub.

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.

Kurangi lompatan jaringan

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.

Konfigurasi kernel Linux

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 socket 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.

```
$ C:\> 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.

```
$ C:\> sudo sysctl -w net.core.busy_poll=50
```

- Konfigurasi status daya CPU (status-C) — Status-C mengontrol tingkat tidur yang mungkin dimasuki inti saat tidak aktif. Anda mungkin ingin mengontrol status-C untuk menyesuaikan sistem Anda untuk latensi versus performa. Dalam keadaan C yang lebih dalam, CPU pada dasarnya “tertidur” 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 mengonfigurasi sistem sehingga tidak menggunakan status C yang lebih dalam. 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 keadaan C yang lebih dalam. Untuk informasi selengkapnya, lihat [Kinerja tinggi dan latensi rendah dengan membatasi status C yang lebih](#) dalam di Panduan Pengguna Amazon Linux 2.

Konfigurasi driver ENA

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.

```
$ C:\> 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.

```
$ C:\> sudo ethtool -C interface adaptive-rx off rx-usecs 0 tx-usecs 0
```

Pertimbangan sistem nitro untuk penyetelan kinerja

Nitro System adalah kumpulan komponen perangkat keras dan perangkat lunak yang dibangun oleh AWS yang memungkinkan performa tinggi, ketersediaan tinggi, dan keamanan tinggi. Sistem Nitro menyediakan kemampuan seperti logam kosong 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 EC2 Nitro. 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 instance berbasis Nitro Anda.

Kartu Nitro menangani semua antarmuka input dan output (I/O), seperti yang diperlukan untuk Virtual Private Clouds (VPC). Untuk semua komponen yang mengirim atau menerima informasi melalui jaringan, kartu Nitro bertindak sebagai perangkat komputasi mandiri untuk lalu lintas I/O 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 tingkat throughput paket per

detik (PPS). Ketika kartu Nitro melakukan evaluasi awal untuk aliran baru, ia 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.

Tingkat 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 mendesain beban kerja Anda.

Bagaimana koneksi dibuat

Ketika koneksi dibuat antara instance berbasis Nitro dan titik akhir lainnya, 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, lihat [Paket yang tidak menggunakan akselerasi perangkat keras](#).

Properti berikut mendefinisikan dua titik akhir dan aliran paket di antara mereka. Kelima sifat ini bersama-sama dikenal sebagai aliran 5-tuple.

- IP sumber
- Port sumber
- IP Tujuan
- Port tujuan
- Protokol komunikasi

Arah aliran paket dikenal sebagai ingress (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 yang 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.

Desain 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 serta desain dan konfigurasi beban kerja aplikasi Anda dapat memengaruhi pemrosesan paket dan tingkat koneksi. Misalnya, jika aplikasi Anda memiliki tingkat pembentukan koneksi yang tinggi, seperti layanan DNS, firewall, atau router virtual, itu 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

Saat Anda mengonfigurasi lalu lintas jaringan untuk instans 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 sebuah

instance melalui satu antarmuka jaringan dan pergi melalui antarmuka jaringan yang berbeda, dapat mengurangi kinerja puncak yang dapat dicapai oleh instans jika arus dilacak. Untuk informasi selengkapnya tentang pelacakan koneksi grup keamanan, koneksi yang tidak dilacak, dan koneksi yang dilacak secara otomatis, lihat. [Pelacakan koneksi grup keamanan](#)

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 tambalan terbaru dan dapat memanfaatkan peningkatan kinerja, seperti fitur jalur akselerasi yang hanya tersedia untuk driver generasi terbaru. Driver sebelumnya tidak mendukung fitur jalur akselerasi.

Untuk memanfaatkan fitur jalur akselerasi, kami sarankan Anda menginstal driver ENA terbaru pada instans Anda.

Instance Linux — ENA Linux driver 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 Windows driver 2.0.0 atau yang lebih baru. Untuk menginstal atau memperbarui driver ENA Windows, lihat [Instal driver Elastic Network Adapter \(ENA\)](#).

Jarak antara titik akhir

Koneksi antara dua instance di Availability Zone yang sama dapat memproses lebih banyak paket per detik daripada koneksi di seluruh Wilayah sebagai akibat dari windowing TCP pada lapisan aplikasi, yang menentukan berapa banyak data yang dapat dalam penerbangan pada waktu tertentu. Jarak yang jauh antar instance meningkatkan latensi dan mengurangi jumlah paket yang dapat diproses oleh titik akhir.

Pertimbangan desain aplikasi

Ada aspek desain dan konfigurasi aplikasi yang dapat memengaruhi 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 oleh sebuah instance di jaringan. Ukuran paket yang lebih kecil dapat meningkatkan laju

proses paket, tetapi ini dapat mengurangi bandwidth maksimum yang dicapai ketika jumlah paket melebihi tunjangan PPS.

Jika ukuran paket melebihi Maximum Transmission Unit (MTU) dari jaringan hop, router di sepanjang jalur mungkin memecahnya. Fragmen paket yang dihasilkan dianggap pengecualian, dan diproses pada tingkat standar (tidak dipercepat). Ini dapat menyebabkan variasi dalam kinerja Anda. Amazon EC2 mendukung frame jumbo 9001 byte, namun tidak semua layanan mendukungnya. Kami menyarankan Anda mengevaluasi topologi Anda ketika Anda mengkonfigurasi MTU.

Pertukaran 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.

Meledak mikro

Micro-bursting terjadi ketika lalu lintas melebihi tunjangan selama periode waktu yang singkat daripada didistribusikan secara merata. Ini biasanya terjadi pada skala mikrodetik.

Misalnya, katakanlah 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 paruh pertama kedua dan tidak meninggalkan apa pun untuk sisa detik. Meskipun Anda mengirim 10Gb dalam jangka waktu 1 detik, tunjangan di paruh pertama detik dapat mengakibatkan paket diantrian atau dijatuhkan.

Anda dapat menggunakan penjadwal jaringan seperti Linux Traffic Control untuk membantu mempercepat throughput Anda dan menghindari menyebabkan paket antrian atau jatuh sebagai akibat dari ledakan mikro.

Jumlah arus

Aliran tunggal 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 dibandingkan dengan 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.

Jumlah antrian Adaptor Jaringan Elastis (ENA)

Secara default, jumlah maksimum antrian ENA dialokasikan ke antarmuka jaringan berdasarkan ukuran dan jenis instans Anda. Mengurangi jumlah antrian dapat mengurangi tingkat PPS maksimum yang dapat dicapai. Sebaiknya gunakan alokasi antrian default untuk performa terbaik.

Untuk Linux, antarmuka jaringan dikonfigurasi dengan maksimum secara default. Untuk aplikasi berdasarkan Data Plane Development Kit (DPDK), kami sarankan Anda mengonfigurasi jumlah antrian maksimum yang tersedia.

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 tingkat pemrosesan paket.

Pelacakan koneksi untuk mempertahankan status

Grup keamanan Anda menggunakan pelacakan koneksi untuk menyimpan informasi tentang lalu lintas ke dan dari instans. Pelacakan koneksi menerapkan aturan terhadap setiap arus lalu lintas jaringan individu untuk menentukan apakah lalu lintas diizinkan atau ditolak. Kartu Nitro menggunakan pelacakan aliran untuk mempertahankan status aliran. Karena semakin banyak aturan kelompok keamanan diterapkan, lebih banyak pekerjaan diperlukan untuk mengevaluasi aliran.

Note

Tidak semua arus lalu lintas jaringan dilacak. Jika aturan grup keamanan dikonfigurasi [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 arus

jaringan Anda. Alur jaringan harus andal memenuhi standar protokol, sesuai dengan perubahan dalam desain VPC, dan paket rute 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 ditangani sebagai pengecualian, dan tidak dapat memanfaatkan akselerasi perangkat keras.

Koneksi menganggur

Ketika koneksi tidak memiliki aktivitas untuk sementara waktu, bahkan jika koneksi belum mencapai batas waktu tunggu, sistem dapat tidak memprioritaskannya. 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 TCP keepalives 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.

Maksimalkan kinerja jaringan pada sistem Nitro Anda

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 meninjau [Pertimbangan](#).

Untuk pertimbangan dan praktik terbaik lainnya untuk konfigurasi instans Anda, lihat:

Instans Linux — [Panduan Praktik Terbaik dan Pengoptimalan Kinerja Driver ENA Linux](#) di GitHub situs web.

Contoh Windows — [Praktik terbaik untuk mengonfigurasi antarmuka jaringan](#).

2. Benchmark beban kerja Anda dengan jumlah alur aktif puncak untuk menentukan dasar kinerja aplikasi Anda. Dengan baseline kinerja, Anda dapat menguji variasi dalam pengaturan atau desain aplikasi untuk memahami pertimbangan mana yang paling berdampak, terutama jika Anda berencana untuk meningkatkan atau meningkatkan skala.

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 mengirim dan menerima instance berada di Availability Zone yang sama atau menggunakan grup penempatan cluster, Anda dapat mengurangi jumlah hop yang perlu diambil 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 atau lebih aliran individu untuk menyebarkan pekerjaan secara merata di seluruh kartu Nitro.

Pantau kinerja pada instance Linux

Anda dapat menggunakan metrik Ethtool pada instans Linux untuk memantau indikator kinerja jaringan instans seperti bandwidth, laju paket, dan pelacakan koneksi. Untuk informasi selengkapnya, lihat [Memantau performa jaringan untuk instans EC2 Anda](#).

Optimalkan kinerja jaringan pada instance Windows

Untuk mencapai kinerja jaringan maksimum pada instance 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 pembongkaran checksum 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 pulang-pergi rata-rata antara instance yang diluncurkan dalam kelompok 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.

Mengonfigurasi afinitas CPU RSS

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. ENA ENI menyediakan hingga delapan antrean 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 jenis instans dengan lebih dari 16 vCPU, kami sarankan Anda menggunakan `Set-NetAdapterRSS` PowerShell cmdlet, yang secara manual mengecualikan prosesor boot (prosesor logis 0 dan 1 ketika hyper-threading diaktifkan) dari konfigurasi RSS untuk semua ENI, untuk mencegah pertengkar dengan berbagai komponen sistem.

Windows memahami fungsi hyper-thread dan akan memastikan antrean RSS dari NIC tunggal selalu ditempatkan pada core fisik yang berbeda. Oleh karena itu, kecuali hyper-threading dinonaktifkan, untuk sepenuhnya mencegah konflik dengan NIC yang lain, sebarkan konfigurasi RSS dari tiap-tiap NIC di antara rentang 16 prosesor logical. `Set-NetAdapterRssCmdlet` memungkinkan Anda untuk menentukan rentang per-NIC prosesor logis yang valid dengan mendefinisikan nilai `BaseProcessorGroup`, `BaseProcessorNumber`, `MaxProcessingGroup`, `MaxProcessorNumber`, dan (opsional) `NumaNode`. Jika tidak ada cukup core fisik menghilangkan konflik antar-NIC sepenuhnya, minimalkan rentang yang tumpang tindih atau kurangi jumlah prosesor logical dalam rentang ENI sesuai dengan beban kerja yang diharapkan dari ENI (dengan kata lain, jaringan admin volume rendah tidak memerlukan banyak antrean 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, jika ada tiga ENI di instans 72 vCPU dengan 2 simpul NUMA yang mengaktifkan hyper-threading, perintah berikut menyebarkan beban jaringan di antara kedua CPU tanpa tumpang tindih serta mencegah penggunaan core 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 instans diubah ukurannya menjadi instans dengan jumlah vCPU yang berbeda, Anda harus mengevaluasi ulang konfigurasi RSS untuk setiap ENI yang diaktifkan. Dokumentasi Microsoft lengkap untuk cmdlet `Set-NetAdapterRss` dapat ditemukan di sini: <https://docs.microsoft.com/en-us/powershell/module/netadapter/set-netadapterrss>.

Catatan khusus untuk beban kerja SQL: Kami juga menyarankan Anda meninjau pengaturan afinitas thread I/O Anda beserta konfigurasi RSS ENI Anda untuk meminimalkan konflik I/O dan jaringan CPU yang sama. Lihat [affinity mask Server Configuration Option](#).

Elastic Fabric Adapter

Elastic Fabric Adapter (EFA) adalah perangkat jaringan yang dapat dipasang pada instans Amazon EC2 Anda untuk mempercepat Komputasi Performa Tinggi (HPC) dan aplikasi machine learning. EFA memungkinkan Anda mencapai kinerja aplikasi kluster HPC lokal, 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. EFA meningkatkan performa komunikasi antarinstans yang sangat penting untuk menskalakan HPC dan aplikasi machine learning. Hal ini dioptimalkan untuk bekerja pada infrastruktur AWS jaringan yang ada dan dapat skala tergantung pada persyaratan aplikasi.

EFA berintegrasi dengan Libfabric 1.7.0 dan yang lebih baru serta mendukung Open MPI 5 dan yang lebih baru serta Intel MPI 2019 Update 5 dan yang lebih baru untuk aplikasi HPC, dan Nvidia Collective Communications Library (NCCL) untuk aplikasi machine learning.

Note

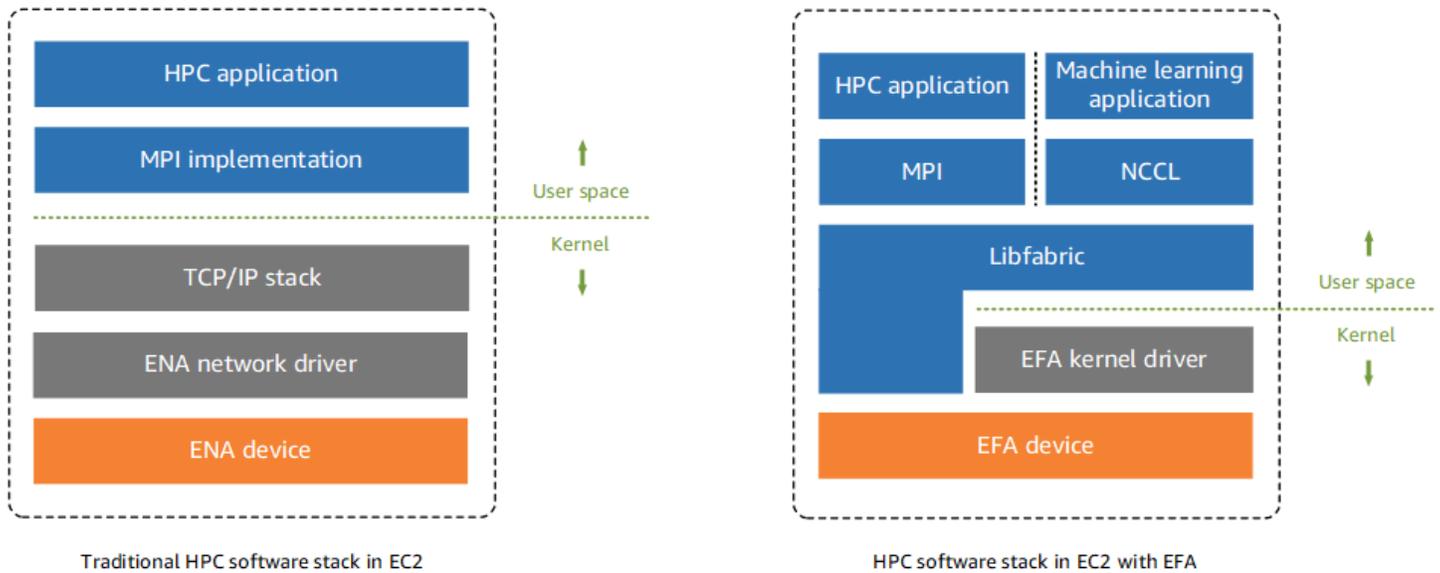
Kapabilitas bypass OS dari EFA tidak didukung di instans Windows. Jika Anda memasang EFA ke instans Windows, instans tersebut berfungsi sebagai Elastic Network Adapter, tanpa kemampuan EFA tambahan.

Daftar Isi

- [Dasar-dasar EFA](#)
- [Antarmuka dan pustaka yang didukung](#)
- [Tipe instans yang didukung](#)
- [Sistem operasi yang didukung](#)
- [Batasan EFA](#)
- [Harga EFA](#)
- [Memulai instans P5 dan EFA](#)
- [Memulai dengan EFA dan MPI](#)
- [Memulai dengan EFA dan NCCL](#)
- [Bekerja dengan EFA](#)
- [Memantau EFA](#)
- [Memverifikasi penginstal EFA menggunakan checksum](#)

Dasar-dasar EFA

EFA adalah Adaptor Jaringan Elastis (ENA) dengan kapabilitas tambahan. EFA menyediakan semua fungsi ENA, dengan fungsi bypass-OS tambahan. Bypass OS adalah model akses yang memungkinkan aplikasi HPC dan machine learning berkomunikasi secara langsung dengan perangkat keras antarmuka jaringan untuk memberikan fungsi transportasi yang berlatensi rendah dan andal.



Secara tradisional, aplikasi HPC menggunakan Message Passing Interface (MPI) untuk menghubungkan transportasi jaringan sistem. Di AWS Cloud, ini berarti bahwa antarmuka aplikasi dengan MPI, yang kemudian menggunakan tumpukan TCP/IP sistem operasi dan driver perangkat ENA untuk mengaktifkan komunikasi jaringan antar instance.

Dengan EFA, aplikasi HPC menggunakan MPI atau NCCL untuk terhubung dengan API Libfabric. API Libfabric memintas kernel sistem operasi dan berkomunikasi secara langsung dengan perangkat ITEFA untuk menempatkan paket pada jaringan. Hal ini mengurangi pengeluaran tambahan dan memungkinkan aplikasi HPC berjalan lebih efisien.

Note

Libfabric adalah komponen inti dari kerangka OpenFabrics Interfaces (OFI), yang mendefinisikan dan mengeksport API ruang pengguna OFI. Untuk informasi lebih lanjut, lihat situs OpenFabrics web [Libfabric](#).

Perbedaan antara EFA dan ENA

Elastic Network Adapters (ENA) menyediakan fitur jaringan IP tradisional yang diperlukan untuk mendukung jaringan VPC. EFA menyediakan semua fitur jaringan IP tradisional yang sama seperti ENA, dan mereka juga mendukung kemampuan bypass OS. Dengan bypass OS, HPC dan aplikasi machine learning dapat melewati kernel sistem operasi dan berkomunikasi secara langsung dengan perangkat EFA.

Antarmuka dan pustaka yang didukung

EFA mendukung antarmuka dan pustaka berikut:

- Open MPI 5 dan yang lebih baru
- Open MPI 4.0 atau lebih baru lebih disukai untuk Graviton
- Pembaruan 5 Intel MPI 2019 dan versi yang lebih baru
- NVIDIA Collective Communications Library (NCCL) 2.4.2 dan yang lebih baru

Tipe instans yang didukung

Tipe instans berikut mendukung EFA:

- Tujuan umum: m5dn.24xlarge m5dn.metal | m5n.24xlarge | m5n.metal | m5zn.12xlarge | m5zn.metal | m6a.48xlarge | m6a.metal | m6i.32xlarge m6i.metal | m6id.32xlarge | m6id.metal | m6idn.32xlarge | m6idn.metal | m6in.32xlarge | m6in.metal | m7a.48xlarge | m7a.metal-48xl m7g.16xlarge | m7g.metal | m7gd.16xlarge | m7gd.metal | m7i.48xlarge | m7i.metal-48xl
- Komputasi dioptimalkan: c5n.9xlarge c5n.18xlarge c5n.metal c6a.48xlarge | c6a.metal | c6gn.16xlarge | c6i.32xlarge | c6i.metal | c6id.32xlarge | c6id.metal | c6in.32xlarge | c6in.metal | c7a.48xlarge | c7a.metal-48xl | c7g.16xlarge | c7g.metal | c7gd.16xlarge | c7gd.metal | c7gn.16xlarge | c7gn.metal | c7i.48xlarge c7i.metal-48xl
- Memori dioptimalkan: r5dn.24xlarge r5dn.metal r5n.24xlarge r5n.metal r6a.48xlarge r6a.metal | r6i.32xlarge | r6i.metal | r6idn.32xlarge | r6idn.metal | r6in.32xlarge r6in.metal | r6id.32xlarge | r6id.metal | r7a.48xlarge | r7a.metal-48xl | r7g.16xlarge | r7g.metal r7gd.16xlarge | r7gd.metal | r7i.48xlarge | r7i.metal-48xl | r7iz.32xlarge r7iz.metal-32xl | u7i-12tb.224xlarge | u7in-16tb.224xlarge | u7in-24tb.224xlarge | u7in-32tb.224xlarge | x2idn.32xlarge x2idn.metal | x2iedn.32xlarge | x2iedn.metal | x2iezn.12xlarge | x2iezn.metal
- Penyimpanan yang dioptimalkan: i3en.12xlarge | i3en.24xlarge | i3en.metal | i4g.16xlarge | i4i.32xlarge | i4i.metal | im4gn.16xlarge
- Komputasi dipercepat: dl1.24xlarge dl2q.24xlarge g4dn.8xlarge g4dn.12xlarge | g4dn.16xlarge | g4dn.metal | g5.8xlarge | g5.12xlarge | g5.16xlarge | g5.24xlarge

g5.48xlarge | g6.8xlarge | g6.12xlarge | g6.16xlarge | g6.24xlarge | g6.48xlarge
| gr6.8xlarge | inf1.24xlarge | p3dn.24xlarge | p4d.24xlarge | p4de.24xlarge |
p5.48xlarge | trn1.32xlarge | trn1n.32xlarge | vt1.24xlarge

- Komputasi berkinerja tinggi: hpc6a.48xlarge hpc6id.32xlarge hpc7a.12xlarge
| hpc7a.24xlarge | | hpc7a.48xlarge | hpc7a.96xlarge | hpc7g.4xlarge |
hpc7g.8xlarge hpc7g.16xlarge

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

Sistem operasi berikut mendukung EFA dengan tipe instans berbasis Intel/AMD x86:

- Amazon Linux 2023
- Amazon Linux 2
- CentOS 7
- RHEL 7, 8, dan 9
- Debian 10 dan 11
- Rocky Linux 8 dan 9
- Ubuntu 20.04 dan 22.04
- SUSE Linux Enterprise 15 SP2 dan yang lebih baru
- OpenSUSE Leap 15.4 dan lebih baru

Note

Ubuntu 20.04 mendukung dukungan peer direct saat digunakan dengan d11.24xlarge instans.

Sistem operasi berikut mendukung EFA dengan tipe instans berbasis Arm (Graviton):

- Amazon Linux 2023
- Amazon Linux 2
- RHEL 8/9 dan Rocky Linux 8/9
- Debian 10 dan 11
- Ubuntu 20.04 dan 22.04
- SUSE Linux Enterprise 15 SP2 dan yang lebih baru

Batasan EFA

EFA memiliki batasan sebagai berikut:

- Semua tipe instans P4d dan P5 mendukung NVIDIA GPUDirect Remote Direct Memory Access (RDMA).
- Lalu lintas EFA antara instans P4D/P4DE/DL1 dan tipe 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.
- Untuk c7g.16xlarge, m7g.16xlarge dan Instans r7g.16xlarge Khusus dan Host Khusus tidak didukung saat EFA dilampirkan.
- Lalu lintas bypass OS EFA terbatas ke satu subnet. Dengan kata lain, lalu lintas EFA tidak dapat dikirim dari satu subnet ke subnet lainnya. Lalu lintas IP normal dari EFA dapat dikirim dari satu subnet ke subnet lainnya.
- Lalu lintas bypass OS EFA tidak dapat diteruskan. Lalu lintas IP normal dari EFA tetap tidak dapat diteruskan.
- EFA harus merupakan anggota grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri.
- EFA tidak didukung pada instance Windows.
- EFA tidak didukung di AWS [Outposts](#).

Harga EFA

EFA tersedia sebagai fitur jaringan Amazon EC2 opsional yang dapat Anda aktifkan pada instans apa pun yang didukung tanpa biaya tambahan.

Memulai instans P5 dan EFA

Instans P5 menghasilkan bandwidth jaringan 3200 Gbps dengan menggunakan beberapa antarmuka EFA. Instans P5 mendukung 32 kartu jaringan. Untuk informasi selengkapnya tentang memulai instans P5, lihat [Memulai instans P5 untuk Linux](#).

Kami menyarankan Anda menentukan antarmuka jaringan EFA tunggal per kartu jaringan. Untuk mengonfigurasi antarmuka ini saat peluncuran, kami merekomendasikan pengaturan berikut:

- Untuk antarmuka jaringan 0, tentukan indeks perangkat 0
- Untuk antarmuka jaringan 1 melalui 31, tentukan indeks perangkat 1

Jika Anda menggunakan konsol Amazon EC2, di Wizard Peluncuran Instans, pilih Edit di bagian Pengaturan jaringan. Perluas Konfigurasi jaringan lanjutan dan pilih Tambahkan antarmuka jaringan untuk menambahkan jumlah antarmuka jaringan yang diperlukan. Untuk setiap antarmuka jaringan, untuk EFA, pilih Aktifkan. Untuk semua antarmuka jaringan, kecuali antarmuka jaringan primer, untuk indeks Perangkat, tentukan 1. Konfigurasi parameter yang tersisa sesuai kebutuhan.

Jika Anda menggunakan AWS CLI, gunakan perintah [run-instance](#), untuk `--network-interfaces`, tentukan jumlah antarmuka jaringan yang diperlukan. Untuk setiap antarmuka jaringan, untuk `InterfaceType`, tentukan `efa`. Untuk antarmuka jaringan primer, untuk `NetworkCardIndex` dan `DeviceIndex` tentukan 0. Untuk antarmuka jaringan yang tersisa, untuk `NetworkCardIndex` tentukan nilai unik dari 1 ke 31, dan untuk `DeviceIndex` tentukan 1.

Contoh cuplikan perintah berikut menunjukkan permintaan dengan 32 antarmuka jaringan EFA.

```
$ 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=efa  
  \  
  "NetworkCardIndex=1,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa  
  \  
  "NetworkCardIndex=2,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa  
  \  
  "
```

```
"NetworkCardIndex=3,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=4,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=5,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=6,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=7,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=8,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=9,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\  
"NetworkCardIndex=10,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=11,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=12,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=13,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=14,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=15,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=16,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  

```

```
"NetworkCardIndex=17,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=18,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=19,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=20,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=21,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=22,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=23,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=24,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=25,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=26,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=27,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=28,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=29,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=30,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\  
"NetworkCardIndex=31,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
```

...

Jika Anda menggunakan template peluncuran, tentukan jumlah antarmuka jaringan yang diperlukan di template peluncuran. Untuk setiap antarmuka jaringan, untuk `InterfaceType`, tentukan `efa`. Untuk antarmuka jaringan primer, untuk `NetworkCardIndex` dan `DeviceIndex` tentukan `0`. Untuk antarmuka jaringan yang tersisa, untuk `NetworkCardIndex` tentukan nilai unik dari 1 ke 31, dan untuk `DeviceIndex` tentukan 1. Cuplikan berikut menunjukkan contoh dengan 3 antarmuka jaringan dari kemungkinan 32 antarmuka jaringan.

```
"NetworkInterfaces":[
{
  "NetworkCardIndex":0,
  "DeviceIndex":0,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 1,
  "DeviceIndex": 1,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 2,
  "DeviceIndex": 1,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
}
...

```

Saat meluncurkan instans P5 dengan lebih dari satu antarmuka jaringan, Anda tidak dapat menetapkan alamat IP publik secara otomatis. Namun, Anda dapat melampirkan alamat IP Elastis ke antarmuka jaringan utama (`NetworkCardIndex=0`, `DeviceIndex=0`) setelah peluncuran untuk konektivitas internet. Baik Ubuntu 20.04 dan yang lebih baru dan Amazon Linux 2 dan yang lebih baru dikonfigurasi untuk menggunakan antarmuka jaringan primer untuk lalu lintas internet ketika instans diluncurkan seperti yang direkomendasikan di atas.

Memulai dengan EFA dan MPI

Tutorial berikut membantu Anda meluncurkan kluster instans yang diaktifkan EFA dan MPI untuk beban kerja HPC. Dalam tutorial ini, Anda akan melakukan langkah-langkah berikut:

Daftar Isi

- [Langkah 1: Siapkan grup keamanan yang diaktifkan EFA](#)
- [Langkah 2: Luncurkan instans sementara](#)
- [Langkah 3: Instal perangkat lunak EFA](#)
- [Langkah 4: \(Opsional\) Mengaktifkan 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: Membuat AMI yang diaktifkan EFA](#)
- [Langkah 10: Meluncurkan instans yang diaktifkan EFA ke dalam grup penempatan kluster](#)
- [Langkah 11: Mengakhiri instans sementara](#)
- [Langkah 12: Mengaktifkan SSH tanpa kata sandi](#)

Langkah 1: Siapkan grup keamanan yang diaktifkan EFA

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 keamanan yang diaktifkan EFA

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 untuk tujuan peluncuran instans Anda yang didukung EFA.
 - 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 tipe 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 instans ini untuk membuat AMI yang diaktifkan EFA sebagai tempat untuk meluncurkan instans Anda yang diaktifkan EFA.

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. 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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. Di bagian Penyimpanan, konfigurasi volume sesuai kebutuhan.
9. Di panel Ringkasan di sebelah kanan, pilih Luncurkan instans.

Langkah 3: Instal perangkat lunak EFA

Instal kernel yang diaktifkan EFA, driver EFA, Libfabric, dan tumpukan Open MPI 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.

Untuk menginstal perangkat lunak EFA

1. Hubungkan ke instans yang Anda luncurkan. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Linux Anda](#).
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, Amazon Linux 2, RHEL 7/8/9, CentOS 7, Linux Rocky 8/9

```
$ sudo yum update -y
```

- Ubuntu 20.04/22.04 dan Debian 10/11

```
$ 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.

```
$ C:\> curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz
```

Anda juga bisa mendapatkan versi terbaru dengan mengganti nomor versi dengan `latest` dalam perintah sebelumnya.

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 AWS Support.

- c. Unduh file tanda tangan dan verifikasi tanda tangan pada file tarball EFA.

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz.sig && gpg --verify ./aws-efa-installer-1.32.0.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 AWS Support.

6. Ekstraksi file dari file `.tar.gz` yang dikompresi dan navigasi ke dalam direktori yang diekstraksi.

```
$ C:\> tar -xf aws-efa-installer-1.32.0.tar.gz && cd aws-efa-installer
```

7. Instal perangkat lunak EFA. Lakukan salah satu langkah berikut sesuai dengan kasus penggunaan Anda.

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 out-of-tree kernel.

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 dan Open MPI 5 sama-sama diinstal secara default. Anda dapat secara opsional menentukan versi Open MPI yang ingin diinstal. Untuk menginstal Open MPI 4 saja, sertakan `--mpi=openmpi4`. Untuk menginstal Open MPI 5 saja, sertakan `--mpi=openmpi5`. Untuk menginstal keduanya, hilangkan opsi `--mpi`.

```
$ C:\> sudo ./efa_installer.sh -y
```

Libfabric diinstal ke `/opt/amazon/efa`. Open MPI 4 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 dan Open MPI 5 sama-sama diinstal secara default. Anda dapat secara opsional menentukan versi Open MPI yang ingin diinstal. Untuk menginstal Open MPI 4 saja, sertakan `--mpi=openmpi4`. Untuk menginstal Open MPI 5 saja, sertakan `--mpi=openmpi5`. Untuk menginstal keduanya, hilangkan opsi `--mpi`.

```
$ C:\> sudo ./efa_installer.sh -y
```

Libfabric diinstal ke `/opt/amazon/efa`. Open MPI 4 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.

```
$ C:\> sudo ./efa_installer.sh -y --minimal
```

8. 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.

Langkah 4: (Opsional) Mengaktifkan Open MPI 5

Note

Lakukan langkah ini hanya jika Anda bermaksud menggunakan Open MPI 5.

Dari EFA 1.30.0, Open MPI 4 dan Open MPI 5 sama-sama diinstal secara default. Atau, Anda dapat memilih untuk menginstal Open MPI 4 atau Open MPI 5 saja.

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.

```
$ C:\> 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 jaringan TCP/IP, yang mungkin mengakibatkan performa 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 berakhir dengan `impi`. Gunakan `libfabric provider` untuk memeriksa dengan Intel MPI menggunakan EFA atau jaringan TCP/IP. 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 (shm) sistem operasi untuk komunikasi intra-simpul, dan menggunakan Libfabric (ofi) hanya untuk komunikasi antar simpul. 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-pttrace.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.

```
$ C:\> 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 [Mengelola perangkat lunak pada instans AL2 Anda](#) di Panduan Pengguna Amazon Linux 2.

Note

Lihat dokumentasi aplikasi HPC Anda untuk petunjuk penginstalan.

Langkah 9: Membuat AMI yang diaktifkan EFA

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat digunakan kembali untuk meluncurkan instans Anda dengan EFA yang diaktifkan.

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: Meluncurkan instans yang diaktifkan EFA ke dalam grup penempatan kluster

Luncurkan instans yang diaktifkan EFA Anda ke dalam grup penempatan kluster menggunakan AMI yang diaktifkan EFA yang Anda buat di Langkah 7, dan grup keamanan yang diaktifkan EFA yang Anda buat di Langkah 1.

Note

- Meluncurkan instans yang diaktifkan EFA ke dalam grup penempatan kluster bukanlah persyaratan mutlak. Namun, kami menyarankan Anda untuk menjalankan instans yang diaktifkan EFA dalam grup penempatan kluster saat instans diluncurkan ke grup dengan latensi rendah di Zona Ketersediaan tunggal.
- Untuk memastikan kapasitas tersedia saat Anda menskalakan instans kluster, Anda dapat membuat Reservasi Kapasitas untuk grup penempatan kluster Anda. Untuk informasi selengkapnya, lihat [Reservasi Kapasitas dalam grup penempatan kluster](#).

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 [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. 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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. (Opsional) Di bagian Penyimpanan, konfigurasi 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 instans, masukkan jumlah instans yang diaktifkan EFA yang ingin Anda luncurkan, lalu pilih Luncurkan instans.

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: 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 kluster, 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

NVIDIA Collective Communications Library (NCCL) adalah perpustakaan rutinitas komunikasi kolektif standar untuk banyak GPU pada simpul tunggal atau beberapa simpul. NCCL dapat digunakan bersama-sama dengan EFA, Libfabric, dan MPI untuk mendukung berbagai beban kerja machine learning. Untuk informasi lebih lanjut, lihat situs web [NCCL](#).

Note

- NCCL dengan EFA hanya didukung dengan instans p3dn.24xlarge, p4d.24xlarge, dan p5.48xlarge.
- Hanya NCCL 2.4.2 dan yang lebih baru yang didukung EFA.

Tutorial berikut membantu Anda meluncurkan kluster instans yang diaktifkan EFA dan NCCL untuk beban kerja machine learning.

- [Gunakan AMI dasar](#)
- [Gunakan AMI AWS Pembelajaran Mendalam](#)

Gunakan AMI dasar

Langkah-langkah berikut membantu Anda memulai dengan Elastic Fabric Adapter menggunakan AMI untuk salah satu [sistem operasi dasar yang didukung](#).

Note

- Hanya tipe instans p3dn.24xlarge, p4d.24xlarge, p5.48xlarge yang didukung.
- Hanya Amazon Linux 2, RHEL 7/8/9, CentOS 7, Rocky Linux 8/9, dan Ubuntu 20.04/22.04 basis AMI yang didukung.

Daftar Isi

- [Langkah 1: Siapkan grup keamanan yang diaktifkan EFA](#)
- [Langkah 2: Luncurkan instans sementara](#)
- [Langkah 3: Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN](#)
- [Langkah 4: Instal GDRCopy](#)
- [Langkah 5: Instal perangkat lunak EFA](#)
- [Langkah 6: Instal NCCL](#)
- [Langkah 7: Instal aws-ofi-nccl plugin](#)
- [Langkah 8: Instal uji NCCL](#)

- [Langkah 9: Uji konfigurasi EFA dan NCCL Anda](#)
- [Langkah 10: Instal aplikasi machine learning Anda](#)
- [Langkah 11: Membuat AMI dengan EFA dan NCCL yang diaktifkan](#)
- [Langkah 12: Mengakhiri instans sementara](#)
- [Langkah 13: Luncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan kluster](#)
- [Langkah 14: Mengaktifkan SSH tanpa kata sandi](#)

Langkah 1: Siapkan grup keamanan yang diaktifkan EFA

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 keamanan yang diaktifkan EFA

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 untuk tujuan peluncuran instans Anda yang didukung EFA.
 - 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 tipe 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 instans ini untuk membuat AMI yang diaktifkan EFA sebagai tempat untuk meluncurkan instans Anda yang diaktifkan EFA.

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 `p3dn.24xlarge`, `p4d.24xlarge`, atau `p5.48xlarge`.
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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. Di bagian Penyimpanan, konfigurasi 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.

Langkah 3: Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN

Amazon Linux 2

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 yum upgrade -y && sudo reboot
```

Hubungkan kembali ke instans Anda setelah boot ulang.

2. Pasang utilitas yang diperlukan untuk memasang driver GPU Nvidia dan toolkit Nvidia CUDA.

```
$ sudo yum groupinstall 'Development Tools' -y
```

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 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. Tambahkan `GRUB_CMDLINE_LINUX="rdblacklist=nouveau"` ke file grub 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. Siapkan repositori yang dibutuhkan

- a. Instal repositori EPEL untuk DKMS dan aktifkan repo opsional apa pun untuk distribusi Linux Anda.

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- b. Instal kunci GPG publik repositori CUDA.

```
$ distribution='rhel7'
```

- c. Siapkan repositori jaringan CUDA dan perbarui cache repositori.

```
$ ARCH=$( /bin/arch ) \
```

```
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \  
&& sudo yum clean expire-cache
```

- d. (Kernel versi 5.10 saja) Lakukan langkah-langkah ini hanya jika Anda menggunakan Amazon Linux 2 dengan kernel versi 5.10. Jika Anda menggunakan Amazon Linux 2 dengan kernel versi 4.12, lewati langkah-langkah ini. Untuk memeriksa versi kernel Anda, jalankan `uname -r`.
 - i. Buat file konfigurasi driver Nvidia bernama `/etc/dkms/nvidia.conf`.

```
$ sudo mkdir -p /etc/dkms \  
&& echo "MAKE[0]=\''make' -j2 module SYSSRC=\${kernel_source_dir} \  
IGNORE_XEN_PRESENCE=1 IGNORE_PREEMPT_RT_PRESENCE=1 IGNORE_CC_MISMATCH=1 \  
CC=/usr/bin/gcc10-gcc\"" | sudo tee /etc/dkms/nvidia.conf
```

- ii. (p4d.24xlarge dan p5.48xlarge saja) Salin file konfigurasi driver Nvidia.

```
$ sudo cp /etc/dkms/nvidia.conf /etc/dkms/nvidia-open.conf
```

6. Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN.

- p3dn.24xlarge

```
$ sudo yum clean all \  
&& sudo yum -y install kmod-nvidia-latest-dkms nvidia-driver-latest-dkms \  
&& sudo yum -y install cuda-drivers-fabricmanager cuda libcuda8-devel
```

- p4d.24xlarge dan p5.48xlarge

```
$ sudo yum clean all \  
&& sudo yum -y install kmod-nvidia-open-dkms nvidia-driver-latest-dkms \  
&& sudo yum -y install cuda-drivers-fabricmanager cuda libcuda8-devel
```

7. Boot ulang dan terhubung kembali ke instans Anda.
8. (p4d.24xlarge dan p5.48xlarge saja) Mulai layanan Nvidia Fabric Manager, dan pastikan bahwa layanan tersebut dimulai secara otomatis saat instans dimulai. Nvidia Fabric Manager diperlukan untuk NV Switch Management.

```
$ sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-fabricmanager
```

9. 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
```

10. 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.

CentOS 7

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 yum upgrade -y && sudo reboot
```

Hubungkan kembali ke instans Anda setelah boot ulang.

2. Pasang utilitas yang diperlukan untuk memasang driver GPU Nvidia dan toolkit Nvidia CUDA.

```
$ sudo yum groupinstall 'Development Tools' -y \
&& sudo yum install -y tar bzip2 make automake pciutils elfutils-libelf-devel
libglvnd-devel iptables firewalld vim bind-utils
```

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 yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(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 grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. Boot ulang dan terhubung kembali ke instans Anda.
5. Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN.
 - a. Instal repositori EPEL untuk DKMS dan aktifkan repo opsional apa pun untuk distribusi Linux Anda.

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- b. Instal kunci GPG publik repositori CUDA.

```
$ distribution='rhel7'
```

- c. Siapkan repositori jaringan CUDA dan perbarui cache repositori.

```
$ ARCH=$( /bin/arch ) \  
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/  
compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \  
&& sudo yum clean expire-cache
```

- d. Instal NVIDIA, driver CUDA dan cuDNN.

```
$ sudo yum clean all \  
&& sudo yum -y install cuda-drivers-fabricmanager cuda libcuda8-devel
```

6. Menyalakan ulang instans dan hubungkan kembali ke instans tersebut.
7. (p4d.24xlarge dan p5.48xlarge saja) Mulai layanan Nvidia Fabric Manager, dan pastikan bahwa layanan tersebut dimulai secara otomatis saat 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.

RHEL 7/8/9 and Rocky Linux 8/9

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 yum upgrade -y && sudo reboot
```

Hubungkan kembali ke instans Anda setelah boot ulang.

2. Pasang utilitas yang diperlukan untuk memasang driver GPU Nvidia dan toolkit Nvidia CUDA.

```
$ sudo yum groupinstall 'Development Tools' -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 yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(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 grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. Boot ulang dan terhubung kembali ke instans Anda.
5. Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN.
 - a. Instal repositori EPEL untuk DKMS dan aktifkan repo opsional apa pun untuk distribusi Linux Anda.

- RHEL 7

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

- RHEL 8 dan Rocky Linux 8/9

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

- RHEL 9

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

- b. Instal kunci GPG publik repositori CUDA.

```
$ distribution=$(. /etc/os-release;echo $ID`rpm -E "%{?rhel}%{?fedora}"`)
```

- c. Siapkan repositori jaringan CUDA dan perbarui cache repositori.

```
$ ARCH=$( /bin/arch ) \  
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \  
&& sudo yum clean expire-cache
```

- d. Instal NVIDIA, driver CUDA dan cuDNN.

```
$ sudo yum clean all \  
$ sudo yum install nvidia-driver nvidia-cuda-toolkit nvidia-cuda-toolkit
```

```
&& sudo yum -y install cuda-drivers-fabricmanager cuda libcuDnn8-devel
```

- Menyalakan ulang instans dan hubungkan kembali ke instans tersebut.
- (p4d.24xlarge dan p5.48xlarge saja) Mulai layanan Nvidia Fabric Manager, dan pastikan bahwa layanan tersebut dimulai secara otomatis saat instans dimulai. Nvidia Fabric Manager diperlukan untuk NV Switch Management.

```
$ sudo systemctl start nvidia-fabricmanager \  
&& sudo systemctl enable nvidia-fabricmanager
```

- 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
```

- 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.

Ubuntu 20.04/22.04

Untuk menginstal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN

- 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 update && 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.
 - `p3dn.24xlarge`

```
$ 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
```

- p4d.24xlarge dan p5.48xlarge

```
$ 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-kernel-open-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.

Langkah 4: Instal GDRCopy

Instal GDRCopy untuk meningkatkan performa Libfabric. Untuk informasi selengkapnya tentang GDRCopy, lihat [repositori GDRCopy](#).

Amazon Linux 2, CentOS 7, RHEL 7/8/9, and Rocky Linux 8/9

Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

```
$ sudo yum -y install dkms rpm-build make check check-devel subunit subunit-devel
```

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. Bangun paket RPM GDRCopy.

```
$ CUDA=/usr/local/cuda ./build-rpm-packages.sh
```

4. Instal paket GDRCopy RPM.

```
$ sudo rpm -Uvh gdrCOPY-kmod-2.4-1dkms.noarch*.rpm \  
&& sudo rpm -Uvh gdrCOPY-2.4-1.x86_64*.rpm \  
&& sudo rpm -Uvh gdrCOPY-devel-2.4-1.noarch*.rpm
```

Ubuntu 20.04/22.04

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. Bangun paket RPM GDRCopy.

```
$ CUDA=/usr/local/cuda ./build-deb-packages.sh
```

4. Instal paket GDRCopy RPM.

```
$ 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
```

Langkah 5: Instal perangkat lunak EFA

Instal kernel yang diaktifkan EFA, driver EFA, Libfabric, dan tumpukan Open MPI yang diperlukan untuk mendukung EFA pada instans sementara Anda.

Untuk menginstal perangkat lunak EFA

1. Hubungkan ke instans yang Anda luncurkan. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Linux Anda](#).

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.

```
$ C:\> curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.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 AWS Support.

- c. Unduh file tanda tangan dan verifikasi tanda tangan pada file tarball EFA.

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.tar.gz.sig
&& gpg --verify ./aws-efa-installer-1.32.0.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 AWS Support.

4. Ekstraksi file dari file `.tar.gz` yang dikompresi dan navigasi ke dalam direktori yang diekstraksi.

```
$ C:\> tar -xf aws-efa-installer-1.32.0.tar.gz && cd aws-efa-installer
```

5. Unduh skrip penginstalan perangkat lunak EFA.

Note

Dari EFA 1.30.0, Open MPI 4 dan Open MPI 5 sama-sama diinstal secara default. Kami sarankan Anda untuk menginstal Open MPI 4 saja, kecuali jika Anda membutuhkan Open MPI 5. Perintah berikut menginstal Open MPI 4 saja. Jika Anda ingin menginstal Open MPI 4 and Open MPI 5, hapus `--mpi=openmpi4`.

```
$ C:\> sudo ./efa_installer.sh -y --mpi=openmpi4
```

Libfabric diinstal di direktori `/opt/amazon/efa`, sedangkan Open MPI diinstal di direktori `/opt/amazon/openmpi`.

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.

```
$ C:\> 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 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/ncc1.git && cd ncc1
```

3. Bangun dan instal NCCL dan tentukan direktori instalasi CUDA.

```
$ sudo make -j src.build CUDA_HOME=/usr/local/cuda
```

Langkah 7: Instal aws-ofi-nccl plugin

aws-ofi-nccl Plugin ini memetakan API transportasi berorientasi koneksi NCCL ke antarmuka Libfabric yang tidak dapat diandalkan tanpa koneksi. Hal ini memungkinkan Anda untuk menggunakan Libfabric sebagai penyedia jaringan saat menjalankan aplikasi berbasis NCCL. Untuk informasi lebih lanjut tentang aws-ofi-nccl plugin, lihat [aws-ofi-nccl repositori](#).

Untuk menginstal aws-ofi-nccl plugin

1. Navigasi ke direktori beranda Anda.

```
$ cd $HOME
```

2. (Amazon Linux 2 dan Ubuntu saja) Instal utilitas yang diperlukan.

- Amazon Linux 2

```
$ sudo yum install hwloc-devel
```

- Ubuntu 20.04

```
$ sudo apt-get install libhwloc-dev
```

3. Unduh file `aws-ofi-nccl` plugin. File dikemas menjadi tarball yang dikompresi (`.tar.gz`).

```
$ wget https://github.com/aws/aws-ofi-nccl/releases/download/v1.9.1-aws/aws-ofi-nccl-1.9.1-aws.tar.gz
```

4. Ekstraksi file dari file `.tar.gz` yang dikompresi dan navigasikan ke dalam direktori yang diekstraksi.

```
$ tar -xf aws-ofi-nccl-1.9.1-aws.tar.gz && cd aws-ofi-nccl-1.9.1-aws
```

5. Untuk menghasilkan file membuat file, jalankan skrip `configure` dan tentukan direktori penginstalan MPI, Libfabric, NCCL, dan CUDA.

```
$ ./configure --prefix=/opt/aws-ofi-nccl --with-mpi=/opt/amazon/openmpi \  
--with-libfabric=/opt/amazon/efa \  
--with-cuda=/usr/local/cuda \  
--enable-platform-aws
```

6. Tambahkan direktori Open MPI ke variabel `PATH`.

```
$ export PATH=/opt/amazon/openmpi/bin/:$PATH
```

7. Instal `aws-ofi-nccl` plugin.

```
$ make && sudo make install
```

Langkah 8: 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, Amazon Linux 2, RHEL, Rocky Linux 8/9, dan CentOS

```
$ export LD_LIBRARY_PATH=/opt/amazon/efa/lib64:$LD_LIBRARY_PATH
```

- Ubuntu

```
$ export LD_LIBRARY_PATH=/opt/amazon/efa/lib:$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
```

Langkah 9: 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 pengujian dan tentukan file host (`--hostfile`) dan jumlah GPU yang akan digunakan (`-n`). Perintah berikut menjalankan uji `all_reduce_perf` pada 8 GPU di instans itu sendiri, dan menentukan variabel lingkungan berikut.

- `FI_EFA_USE_DEVICE_RDMA=1`—(hanya `p4d.24xlarge`) menggunakan fungsi RDMA perangkat untuk transfer satu sisi dan dua sisi.
- `NCCL_DEBUG=INFO`—memungkinkan output debug terperinci. Anda juga dapat menentukan `VERSION` untuk mencetak versi NCCL saja di awal pengujian, atau `WARN` untuk menerima pesan kesalahan saja.

Untuk informasi lebih lanjut mengenai argumen uji NCCL, lihat [Tes NCCL README](#) dalam repositori `nccl-tests` resmi.

- `p3dn.24xlarge`

```
$ /opt/amazon/openmpi/bin/mpirun \
  -x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-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
```

- `p4d.24xlarge` dan `p5.48xlarge`

```
$ /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/aws-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-24x1-topo.xml
```

Langkah 10: 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 [Mengelola perangkat lunak di instans Amazon Linux 2 Anda](#).

Note

Lihat dokumentasi aplikasi machine learning Anda untuk petunjuk penginstalan.

Langkah 11: Membuat AMI dengan EFA dan NCCL yang diaktifkan

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat digunakan kembali untuk meluncurkan instans Anda dengan EFA yang diaktifkan.

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 12: 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 13: Luncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan klaster

Luncurkan instans yang diaktifkan EFA dan NCCL Anda ke dalam grup penempatan klaster menggunakan AMI yang diaktifkan EFA dan grup keamanan yang diaktifkan EFA yang Anda buat sebelumnya.

Note

- Meluncurkan instans yang diaktifkan EFA ke dalam grup penempatan klaster bukanlah persyaratan mutlak. Namun, kami menyarankan Anda untuk menjalankan instans yang diaktifkan EFA dalam grup penempatan klaster saat instans diluncurkan ke grup dengan 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 [Reservasi Kapasitas dalam grup penempatan klaster](#).

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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. (Opsional) Di bagian Penyimpanan, konfigurasi 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 instans, masukkan jumlah instans yang diaktifkan EFA yang ingin Anda luncurkan, lalu pilih Luncurkan instans.

Old console

Untuk meluncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan klaster

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: Konfigurasi Detail Instans.
5. Pada halaman Konfigurasi Detail Instans, lakukan langkah berikut:
 - a. Untuk Jumlah instans, masukkan jumlah instans yang diaktifkan EFA dan NCCL 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: Konfigurasi 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 Tinjau Peluncuran Instans, tinjau pengaturannya, lalu pilih Luncurkan untuk memilih pasangan kunci dan meluncurkan instans Anda.

Langkah 14: 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 kluster, 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.

Gunakan AMI AWS Pembelajaran Mendalam

Langkah-langkah berikut membantu Anda memulai dengan salah satu AMI Pembelajaran AWS Mendalam berikut:

- AMI Deep Learning (Amazon Linux 2)
- AMI Deep Learning (Ubuntu 20.04)

Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS Deep Learning AMI](#) .

 Note

Hanya tipe instans p3dn.24xlarge dan p4d.24xlarge yang didukung.

Daftar Isi

- [Langkah 1: Siapkan grup keamanan yang diaktifkan EFA](#)
- [Langkah 2: Luncurkan instans sementara](#)
- [Langkah 3: Uji konfigurasi EFA dan NCCL Anda](#)
- [Langkah 4: Instal aplikasi machine learning Anda](#)
- [Langkah 5: Membuat AMI dengan EFA dan NCCL yang diaktifkan](#)
- [Langkah 6: Mengakhiri instans sementara](#)
- [Langkah 7: Luncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan kluster](#)
- [Langkah 8: Mengaktifkan SSH tanpa kata sandi](#)

Langkah 1: Siapkan grup keamanan yang diaktifkan EFA

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 keamanan yang diaktifkan EFA

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 untuk tujuan peluncuran instans Anda yang didukung EFA.
 - 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 tipe 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 instans ini untuk membuat AMI yang diaktifkan EFA sebagai tempat untuk meluncurkan instans Anda yang diaktifkan EFA.

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 Deep Learning AWS Versi 25.0 yang didukung atau yang lebih baru.
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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. Di bagian Penyimpanan, konfigurasi 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.

Langkah 3: 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 pengujian dan tentukan file host (`--hostfile`) dan jumlah GPU yang akan digunakan (`-n`). Perintah berikut menjalankan uji `all_reduce_perf` pada 8 GPU di instans itu sendiri, dan menentukan variabel lingkungan berikut.
 - `FI_EFA_USE_DEVICE_RDMA=1`—(hanya `p4d.24xlarge`) menggunakan fungsi RDMA perangkat untuk transfer satu sisi dan dua sisi.
 - `NCCL_DEBUG=INFO`—memungkinkan output debug terperinci. Anda juga dapat menentukan `VERSION` untuk mencetak versi NCCL saja di awal pengujian, atau `WARN` untuk menerima pesan kesalahan saja.

Untuk informasi lebih lanjut mengenai argumen uji NCCL, lihat [Tes NCCL README](#) dalam repositori `nccl-tests` resmi.

- `p3dn.24xlarge`

```
$ /opt/amazon/openmpi/bin/mpirun \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-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
```

- p4d.24xlarge

```
$ /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/aws-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-24x1-topo.xml
```

Langkah 4: 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 [Mengelola perangkat lunak di instans Amazon Linux 2 Anda](#).

Note

Lihat dokumentasi aplikasi machine learning Anda untuk petunjuk penginstalan.

Langkah 5: Membuat AMI dengan EFA dan NCCL yang diaktifkan

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat digunakan kembali untuk meluncurkan instans Anda dengan EFA yang diaktifkan.

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 6: 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 7: Luncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan klaster

Luncurkan instans yang diaktifkan EFA dan NCCL Anda ke dalam grup penempatan klaster menggunakan AMI yang diaktifkan EFA dan grup keamanan yang diaktifkan EFA yang Anda buat sebelumnya.

Note

- Meluncurkan instans yang diaktifkan EFA ke dalam grup penempatan klaster bukanlah persyaratan mutlak. Namun, kami menyarankan Anda untuk menjalankan instans yang

diaktifkan EFA dalam grup penempatan kluster saat instans diluncurkan ke grup dengan latensi rendah di Zona Ketersediaan tunggal.

- Untuk memastikan kapasitas tersedia saat Anda menskalakan instans kluster, Anda dapat membuat Reservasi Kapasitas untuk grup penempatan kluster Anda. Untuk informasi selengkapnya, lihat [Reservasi Kapasitas dalam grup penempatan kluster](#).

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, dan untuk Elastic Fabric Adapter, pilih Aktifkan.
8. (Opsional) Di bagian Penyimpanan, konfigurasi volume sesuai kebutuhan.
9. Di bagian Detail lanjutan, untuk nama grup Penempatan, pilih grup penempatan kluster untuk meluncurkan instans. Jika Anda perlu membuat grup penempatan kluster baru, pilih Buat grup penempatan baru.
10. Di panel Ringkasan di sebelah kanan, untuk Jumlah instans, masukkan jumlah instans yang diaktifkan EFA yang ingin Anda luncurkan, lalu pilih Luncurkan instans.

Old console

Untuk meluncurkan instans yang diaktifkan EFA dan NCCL ke dalam grup penempatan klaster

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: Konfigurasi Detail Instans.
5. Pada halaman Konfigurasi Detail Instans, lakukan langkah berikut:
 - a. Untuk Jumlah instans, masukkan jumlah instans yang diaktifkan EFA dan NCCL 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: Konfigurasi 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 Tinjau Peluncuran Instans, tinjau pengaturannya, lalu pilih Luncurkan untuk memilih pasangan kunci dan meluncurkan instans Anda.

Langkah 8: Mengaktifkan SSH tanpa kata sandi

Agar aplikasi Anda dapat berjalan di semua instans dalam kluster, 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 kluster adalah simpul anggota.

Untuk mengaktifkan SSH tanpa kata sandi antar instans dalam kluster

1. Pilih satu instans dalam kluster 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 kluster, 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.

Bekerja dengan EFA

Anda dapat membuat, menggunakan, dan mengelola EFA seperti antarmuka jaringan elastis lainnya di Amazon EC2. Namun, tidak seperti antarmuka jaringan elastis, EFA tidak dapat dipasangkan atau dilepaskan dari suatu instans dalam keadaan berjalan.

Persyaratan EFA

Untuk menggunakan EFA, Anda harus melakukan hal berikut:

- Pilih salah satu [tipe instans yang didukung](#).
- Gunakan AMI untuk salah satu [sistem operasi yang didukung](#).
- Instal komponen perangkat lunak EFA. Untuk informasi lebih lanjut, lihat [Langkah 3: Instal perangkat lunak EFA](#) dan [Langkah 5: \(Opsional\) Instal Intel MPI](#).
- EFA menggunakan 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 keamanan yang diaktifkan EFA](#).

Daftar Isi

- [Membuat AMI](#)
- [Memasang EFA ke instans yang dihentikan](#)
- [Memasang EFA saat meluncurkan instans](#)
- [Menambahkan EFA ke templat peluncuran](#)
- [Mengelola alamat IP untuk EFA](#)
- [Mengubah grup keamanan untuk EFA](#)
- [Melepas EFA](#)
- [Melihat EFA](#)
- [Menghapus EFA](#)

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.

Untuk membuat EFA menggunakan konsol

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. Untuk Deskripsi, masukkan nama deskriptif untuk EFA.
5. Untuk Subnet, pilih subnet untuk membuat EFA.
6. Untuk IP Privat, masukkan alamat IPv4 privat utama. Jika Anda tidak menentukan alamat IPv4, kami memilih alamat IPv4 privat dari dalam subnet yang dipilih.
7. (Hanya IPv6) Jika Anda memilih subnet yang terkait blok CIDR IPv6, Anda dapat secara opsional menentukan alamat IPv6 di bidang IP IPv6.
8. Untuk Grup keamanan, pilih satu atau beberapa grup keamanan.
9. Untuk EFA, pilih Diaktifkan.
10. Pilih Ya, Buat.

Untuk membuat EFA baru menggunakan AWS CLI

Gunakan perintah [create-network-interface](#) dan untuk `interface-type`, sebutkan efa, seperti ditunjukkan dalam contoh berikut.

```
aws ec2 create-network-interface --subnet-id subnet-01234567890 --  
description example_efa --interface-type 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 [Melampirkan antarmuka jaringan ke sebuah instans](#).

Memasang EFA saat meluncurkan instans

Untuk memasang EFA yang sudah ada saat meluncurkan instans (AWS CLI)

Gunakan perintah [run-instance](#) dan untuk NetworkInterfaceId, tentukan ID EFA, seperti yang ditunjukkan pada contoh berikut.

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-type c5n.18xlarge --key-name my_key_pair --network-interfaces DeviceIndex=0,NetworkInterfaceId=efa_id,Groups=sg_id,SubnetId=subnet_id
```

Untuk memasang EFA baru saat meluncurkan instans (AWS CLI)

Gunakan perintah [run-instance](#) dan for InterfaceType, tentukan *efa*, seperti yang ditunjukkan pada contoh berikut.

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-type c5n.18xlarge --key-name my_key_pair --network-interfaces DeviceIndex=0,InterfaceType=efa,Groups=sg_id,SubnetId=subnet_id
```

Menambahkan EFA ke templat peluncuran

Anda dapat membuat templat peluncuran yang berisi informasi konfigurasi yang diperlukan untuk meluncurkan instans yang diaktifkan EFA. Untuk membuat templat peluncuran yang diaktifkan EFA, buat templat peluncuran baru dan tentukan tipe instans yang didukung, AMI yang diaktifkan EFA Anda, dan grup keamanan yang diaktifkan EFA. Untuk informasi selengkapnya, lihat [Memulai dengan EFA dan MPI](#).

Anda dapat memanfaatkan templat peluncuran untuk meluncurkan instans yang diaktifkan EFA dengan layanan AWS lainnya, seperti [AWS Batch](#) atau [AWS ParallelCluster](#).

Untuk informasi lebih lanjut tentang membuat templat peluncuran, lihat [Membuat templat peluncuran](#).

Mengelola alamat IP untuk EFA

Anda dapat mengubah alamat IP yang terkait dengan EFA. Jika Anda memiliki alamat IP Elastis, Anda dapat menghubungkannya dengan ITEFA. Jika EFA Anda disediakan dalam subnet yang memiliki Blok CIDR IPv6 terkait, Anda dapat menetapkan satu atau lebih alamat IPv6 untuk EFA.

Anda menetapkan Elastic IP (IPv4) dan alamat IPv6 ke EFA dalam cara yang sama dengan cara Anda menetapkan alamat IP ke antarmuka jaringan elastis. Untuk informasi selengkapnya, lihat [Mengelola alamat IP](#).

Mengubah grup keamanan untuk EFA

Anda dapat mengubah grup keamanan yang terkait dengan EFA. Untuk mengaktifkan fungsi bypass OS, EFA harus merupakan anggota grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri.

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 lebih lanjut, lihat [Mengubah grup keamanan](#).

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 [Melepaskan antarmuka jaringan dari sebuah instans](#).

Melihat EFA

Anda dapat melihat semua EFA di akun Anda.

Anda melihat EFA dengan cara yang sama seperti melihat antarmuka jaringan elastis. Untuk informasi selengkapnya, lihat [Melihat detail tentang 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](#).

Memantau EFA

Anda dapat menggunakan fitur-fitur berikut untuk memantau performa Elastic Fabric Adapters Anda.

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

Amazon CloudWatch menyediakan metrik yang memungkinkan Anda memantau EFA Anda secara real time. Anda dapat mengumpulkan dan melacak metrik, membuat dasbor yang disesuaikan, dan mengatur alarm yang memberi tahu Anda atau mengambil tindakan saat metrik tertentu mencapai ambang batas yang ditentukan. Untuk informasi selengkapnya, lihat [Pantau instans Anda menggunakan CloudWatch](#).

Memverifikasi penginstal EFA menggunakan checksum

Secara opsional, Anda dapat memverifikasi tarball EFA (file `.tar.gz`) 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 AWS Support.

Misalnya, perintah berikut memverifikasi tarball EFA 1.9.4 menggunakan checksum SHA256.

```
$ sha256sum aws-efa-installer-1.9.4.tar.gz
```

```
1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14 aws-efa-  
installer-1.9.4.tar.gz
```

Tabel berikut mencantumkan checksum untuk versi EFA terbaru.

Versi	Unduh URL	Checksum
EFA 1.32.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.32.0.tar.gz	MD5: db8d65cc028d8d08b5 a9f2d88881c1b1 SHA256: 5f7233760be57f6fee 6de8c09acbfbf59238

Versi	Unduh URL	Checksum
		de848e06048dc54d15 6ef578fc66
EFA 1.31.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.31.0.tar.gz	MD5: 856352f12bef2ccbad cd75e35aa52aaf SHA256: 943325bd37902a4300 ac9e5715163537d56e cb4e7b87b37827c3e5 47aa1897bf
EFA 1.30.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.30.0.tar.gz	MD5: 31f48e1a47fe93ede8 ebd273fb747358 SHA256: 876ab9403e07a0c3c9 1a1a34685a52eced89 0ae052df94857f6081 c5f6c78a0a
EFA 1.29.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.29.1.tar.gz	MD5: e1872ca815d752c1d7 c2b5c175e52a16 SHA256: 178b263b8c25845b63 dc93b25bcdff5870df 5204ec509af26f43e8 d283488744
EFA 1.29.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.29.0.tar.gz	MD5: 39d06a002154d94cd9 82ed348133f385 SHA256: 836655f87015547e73 3e7d9f7c760e4e2469 7f8bbc261bb5f3560a bd4206bc36

Versi	Unduh URL	Checksum
EFA 1.28.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.28.0.tar.gz	MD5: 9dc13b744666582260 5e66febe074035 SHA256: 2e625d2d6d3e073b51 78e8e861891273d896 b66d03cb1a32244fd5 6789f1c435
EFA 1.27.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.27.0.tar.gz	MD5: 98bfb515ea3e8d93f5 54020f3837fa15 SHA256: 1d49a97b0bf8d964d9 1652a79ac851f2550e 33a5bf9d0cf86ec935 7ff6579aa3
EFA 1.26.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.26.1.tar.gz	MD5: 884e74671fdef47255 01f7cd2d451d0c SHA256: c616994c924f54ebfa bfab32b7fe8ac56947 fae00a0ff453d975e2 98d174fc96
EFA 1.26.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.26.0.tar.gz	MD5: f8839f12ff2e3b9ba0 9ae8a82b30e663 SHA256: bc1abc1f76e97d204d 3755d2a9ca307fc423 e51c63141f798c2f15 be3715aa11

Versi	Unduh URL	Checksum
EFA 1.25.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.25.1.tar.gz	MD5: 6d876b894547847a45 bb8854d4431f18 SHA256: d2abc553d22b89a4ce 92882052c1fa6de450 d3a801fe005da718b7 d4b9602b06
EFA 1.25.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.25.0.tar.gz	MD5: 1993836ca749596051 da04694ea0d00c SHA256: 98b7b26ce031a2d6a9 3de2297cc71b03af64 7194866369ca53b60d 82d45ad342
EFA 1.24.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.24.1.tar.gz	MD5: 211b249f39d53086f3 cb0c07665f4e6f SHA256: 120cfeec233af09556 23ac7133b674143329 f9561a9a8193e47306 0f596aec62
EFA 1.24.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.24.0.tar.gz	MD5: 7afe0187951e2dd2c9 cc4b572e62f924 SHA256: 878623f819a0d9099d 76ecd41cf4f569d4c3 aac0c9bb7ba9536347 c50b6bf88e

Versi	Unduh URL	Checksum
EFA 1.23.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.23.1.tar.gz	MD5: 22491e114b6ee7160a 8290145dca0c28 SHA256: 5ca848d8e0ff4d1571 cd443c36f8d27c8cdf 2a0c97e9068ebf000c 303fc40797
EFA 1.23.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.23.0.tar.gz	MD5: 38a6d7c1861f5038db a4e441ca7683ca SHA256: 555d497a60f22e3857 fdeb3dfc53aa86d059 26023c68c916d15d2d c3df6525bd
EFA 1.22.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.22.1.tar.gz	MD5: 600c0ad7cdbc06e8e8 46cb763f92901b SHA256: f90f3d5f59c031b9a9 64466b5401e86fd042 9272408f6c207c3f90 48254e9665
EFA 1.22.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.22.0.tar.gz	MD5: 8f100c93dc8ab519c2 aeb5dab89e98f8 SHA256: f329e7d54a86a03ea5 1da6ea9a5b68fb354f bae4a57a02f9592e21 fce431dc3a

Versi	Unduh URL	Checksum
EFA 1.21.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.21.0.tar.gz	MD5: 959ccc3a4347461909 ec02ed3ba7c372 SHA256: c64e6ca34ccfc3ebe8 e82d08899ae8442b3e f552541cf5429c43d1 1a04333050
EFA 1.20.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.20.0.tar.gz	MD5: 7ebfbb8e85f1b94709 df4ab3db47913b SHA256: aeefd2681ffd5c4c63 1d1502867db5b83162 1d6eb85b61fe3ec80d f983d1dcf0
EFA 1.19.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.19.0.tar.gz	MD5: 2fd45324953347ec55 18da7e3fefa0ec SHA256: 99b77821b9e72c8dea 015cc92c96193e8db3 07deee05b91a58094c c331f16709
EFA 1.18.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.18.0.tar.gz	MD5: fc2571a72f5d3c7b7b 576ce2de38d91e SHA256: acb18a0808aedb9a5e 485f1469225b9ac97f 21db9af78e4cd69397 00debe1cb6

Versi	Unduh URL	Checksum
EFA 1.17.3	https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.3.tar.gz	MD5: 0517df4a190356ab55 9235147174cafd SHA256: 5130998b0d2883bbae 189b21ab215ecbc1b0 1ae0231659a9b4a17b 0a33ebc6ca
EFA 1.17.2	https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.2.tar.gz	MD5: a329dedab53c4832df 218a24449f4c9a SHA256: bca1fdde8b32b00346 e175e597ffab32a09a 08ee9ab136875fb382 83cc4cd099
EFA 1.17.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.1.tar.gz	MD5: 733ae2cfc9d14b5201 7eaf0a2ab6b0ff SHA256: f29322640a88ae9279 805993cb836276ea24 0623820848463ca686 c8ce02136f
EFA 1.17.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.17.0.tar.gz	MD5: d430fc841563c11c38 05c5f82a4746b1 SHA256: 75ab0cee4fb6bd3888 9dce313183f5d3a83b d233e0a6ef6205d835 2821ea901d

Versi	Unduh URL	Checksum
EFA 1.16.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.16.0.tar.gz	MD5: 399548d3b0d2e812d7 4dd67937b696b4 SHA256: cecec36495a1bc6fdc 82f97761a541e4fb6c 9a3cbf3cfcb145acf2 5ea5dbd45b
EFA 1.15.2	https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.2.tar.gz	MD5: 955fea580d5170b058 23d51acde7ca21 SHA256: 84df4fbc1b3741b6c0 73176287789a601a58 9313accc8e6653434e 8d4c20bd49
EFA 1.15.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.1.tar.gz	MD5: c4610267039f72bbe4 e35d7bf53519bc SHA256: be871781a1b9a15fca 342a9d169219260069 942a8bda7a8ad06d4b aeb5e2efd7
EFA 1.15.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.15.0.tar.gz	MD5: 9861694e1cc00d884f adac07d22898be SHA256: b329862dd5729d2d09 8d0507fb486bf859d7 c70ce18b61c3029822 34a3a5c88f

Versi	Unduh URL	Checksum
EFA 1.14.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.14.1.tar.gz	MD5: 50ba56397d359e5787 2fde1f74d4168a SHA256: c7b1b48e86fe4b3eaa 4299d3600930919c4f e6d88cc6e2c7e4a408 a3f16452c7
EFA 1.14.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.14.0.tar.gz	MD5: 40805e7fd842c36ece cb9fd7f921b1ae SHA256: 662d62c12de85116df 33780d40e0533ef7da d92709f4f613907475 a7a1b60a97
EFA 1.13.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.13.0.tar.gz	MD5: c91d16556f4fd53bec adbb345828221e SHA256: ad6705eb23a3fce44a f3afc0f76430915956 53a723ad0374084f4f 2b715192e1
EFA 1.12.3	https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.3.tar.gz	MD5: 818aee81f097918cfa ebd724eddea678 SHA256: 2c225321824788b8ca 3fbc118207b944cdb0 96b847e1e0d1d853ef 2f0d727172

Versi	Unduh URL	Checksum
EFA 1.12.2	https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.2.tar.gz	MD5: 956bb1fc5ae0d6f0f8 7d2e481d49fccf SHA256: 083a868a2c212a5a4f cf3e4d732b685ce39c ceb3ca7e5d50d0b74e 7788d06259
EFA 1.12.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.1.tar.gz	MD5: f5bfe52779df435188 b0a2874d0633ea SHA256: 5665795c2b4f09d5f3 f767506d4d4c429695 b36d4a17e5758b27f0 33aee58900
EFA 1.12.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.12.0.tar.gz	MD5: d6c6b49fafb39b7702 97e1cc44fe68a6 SHA256: 28256c57e9ecc0b077 8b41c1f777a9982b4e 8eae782343dfe12460 79933dca59
EFA 1.11.2	https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.2.tar.gz	MD5: 2376cf18d1353a4551 e35c33d269c404 SHA256: a25786f98a3628f7f5 4f7f74ee2b39bc6734 ea9374720507d37d3e 8bf8ee1371

Versi	Unduh URL	Checksum
EFA 1.11.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.1.tar.gz	MD5: 026b0d9a0a48780cc7 406bd51997b1c0 SHA256: 6cb04baf5ffc58ddf3 19e956b5461289199c 8dd805fe216f8f9ab8 d102f6d02a
EFA 1.11.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.11.0.tar.gz	MD5: 7d9058e010ad65bf2e 14259214a36949 SHA256: 7891f6d45ae33e8221 89511c4ea1d14c9d54 d000f6696f97be54e9 15ce2c9dfa
EFA 1.10.1	https://efa-installer.amazonaws.com/ aws-efa-installer-1.10.1.tar.gz	MD5: 78521d3d668be22976 f46c6fecc7b730 SHA256: 61564582de7320b21d e319f532c3a677d26c c46785378eb3b95c63 6506b9bcb4
EFA 1.10.0	https://efa-installer.amazonaws.com/ aws-efa-installer-1.10.0.tar.gz	MD5: 46f73f5a7afe41b4bb 918c81888fef9a9 SHA256: 136612f96f2a085a7d 98296da0afb6fa807b 38142e2fc0c548fa98 6c41186282

Versi	Unduh URL	Checksum
EFA 1.9.5	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.5.tar.gz	MD5: 95edb8a209c18ba8d2 50409846eb6ef4 SHA256: a4343308d7ea4dc943 ccc21bcebed913e886 8e59bf2ac93599c61 a7c87d7d25
EFA 1.9.4	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.4.tar.gz	MD5: f26dd5c350422c1a98 5e35947fa5aa28 SHA256: 1009b5182693490d90 8ef0ed2c1dd4f813cc 310a5d2062ce9619c4 c12b5a7f14
EFA 1.9.3	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.3.tar.gz	MD5: 95755765a097802d3e 6d5018d1a5d3d6 SHA256: 46ce732d6f3fcc9edf 6a6e9f9df0ad136054 328e24675567f7029e dab90c68f1
EFA 1.8.4	https://efa-installer.amazonaws.com/aws-efa-installer-1.8.4.tar.gz	MD5: 85d594c41e831afc6c 9305263140457e SHA256: 0d974655a09b213d78 59e658965e56dc4f23 a0eee2dc44bb41b6d0 39cc5bab45

Topologi instans Amazon EC2

Menjelaskan topologi instance Anda memberikan tampilan hierarkis dari kedekatan relatif antar instance. Anda dapat menggunakan informasi ini untuk mengelola infrastruktur komputasi kinerja

tinggi (HPC) dan pembelajaran mesin (ML) dalam skala besar, sambil mengoptimalkan penempatan kerja. Pekerjaan HPC dan ML sensitif terhadap latensi dan throughput. Anda dapat menggunakan topologi instans untuk mendeteksi lokasi instans Anda, dan kemudian menggunakan informasi ini untuk mengoptimalkan pekerjaan HPC dan ML dengan menjalankannya pada instans yang secara fisik lebih dekat satu sama lain.

Anda dapat menggunakan topologi instance untuk mendeteksi lokasi instance yang ada, tetapi Anda tidak dapat menggunakannya untuk memilih meluncurkan instance baru secara fisik dekat dengan instance yang sudah ada. Untuk mempengaruhi penempatan instance, Anda dapat menggunakan [Reservasi Kapasitas dalam grup penempatan klaster](#).

Harga

Tidak ada biaya tambahan untuk menggambarkan topologi instans Anda.

Daftar Isi

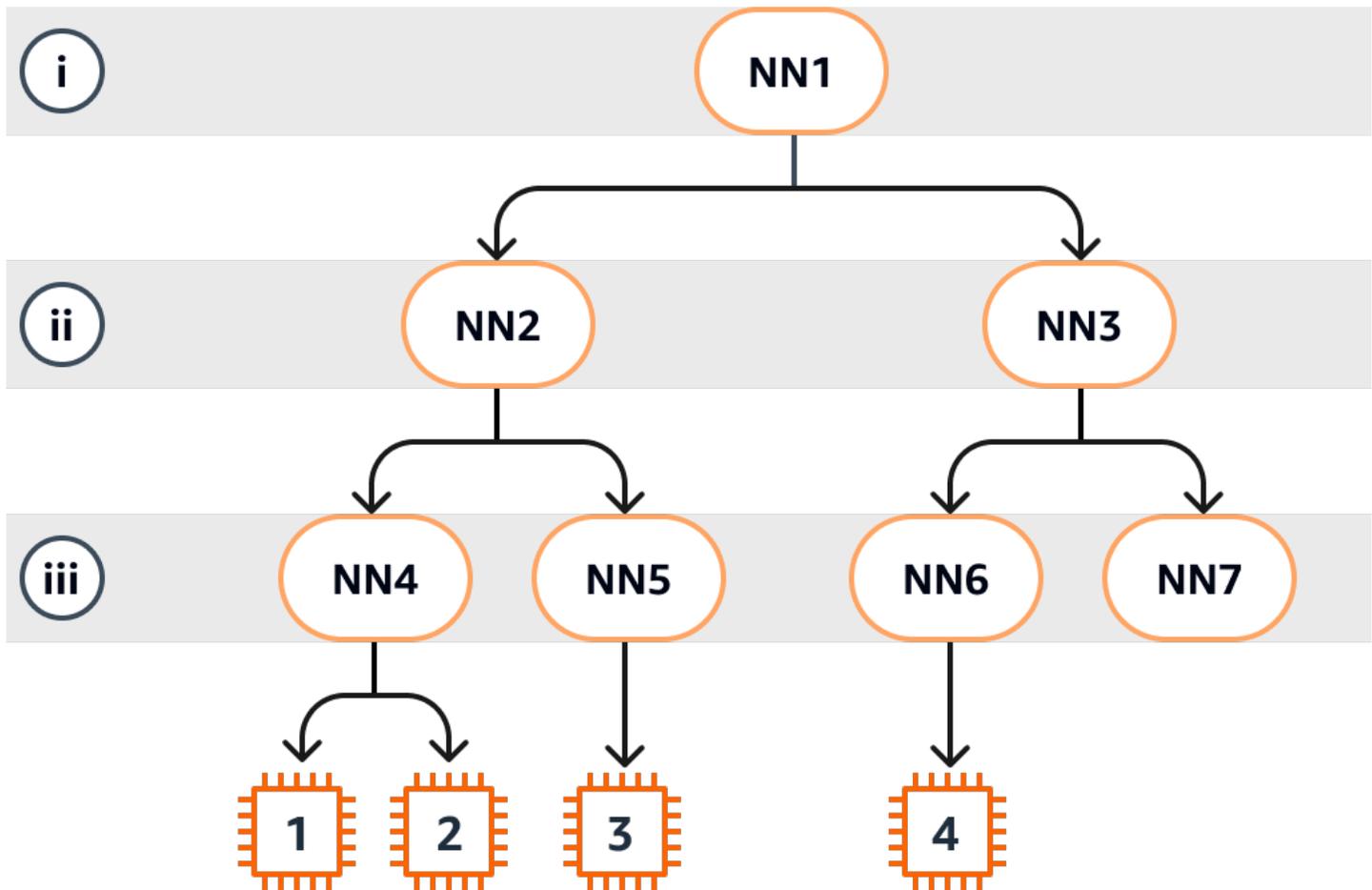
- [Cara kerja topologi instance](#)
- [Prasyarat misalnya topologi](#)
- [Contoh topologi instans Amazon EC2](#)

Cara kerja topologi instance

Setiap instans EC2 terhubung ke set simpul. Sebuah set node terdiri dari tiga node jaringan, dengan masing-masing node mewakili lapisan yang berbeda dalam AWS jaringan. Lapisan jaringan diatur dalam hierarki 3 atau lebih lapisan. Set simpul menyediakan tampilan top-down dari hierarki ini, dengan lapisan bawah terhubung paling dekat dengan sebuah instans.

Informasi tentang set node disebut topologi instance.

Diagram berikut memberikan representasi visual yang dapat Anda gunakan untuk memahami topologi instance. Node jaringan diidentifikasi sebagai NN1 - NN7. Angka i, ii, dan iii mengidentifikasi lapisan jaringan. Angka 1, 2, 3, dan 4 mengidentifikasi instans EC2. Contoh terhubung ke node di lapisan bawah, diidentifikasi oleh iii. 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 output

Anda mendapatkan informasi topologi instance menggunakan API [DescribeInstanceTopology](#). Output memberikan pandangan hierarkis dari topologi jaringan yang mendasari 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` menggambarkan rangkaian simpul jaringan dari sebuah instans.
- 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-111111111example` dan `i-222222222example` terletak paling dekat satu sama lain dibandingkan dengan instans lain dalam contoh ini karena mereka memiliki simpul jaringan yang sama `nn-444444444example` di lapisan bawah.

```
{
  "Instances": [
    {
      "InstanceId": "i-111111111example", //Corresponds to instance 1
      "InstanceType": "p4d.24xlarge",
      "GroupName": "ML-group",
      "NetworkNodes": [
        "nn-111111111example", //Corresponds to NN1 in layer i
        "nn-222222222example", //Corresponds to NN2 in layer ii
        "nn-444444444example" //Corresponds to NN4 in layer iii -
        bottom layer, connected to the instance
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
    {
      "InstanceId": "i-222222222example", //Corresponds to instance 2
      "InstanceType": "p4d.24xlarge",
      "NetworkNodes": [
        "nn-111111111example", //Corresponds to NN1 - layer i
        "nn-222222222example", //Corresponds to NN2 - layer ii
        "nn-444444444example" //Corresponds to NN4 - layer iii -
        connected to instance
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
    {
      "InstanceId": "i-333333333example", //Corresponds to instance 3
      "InstanceType": "trn1.32xlarge",
      "NetworkNodes": [
        "nn-111111111example", //Corresponds to NN1 - layer i
        "nn-222222222example", //Corresponds to NN2 - layer ii
        "nn-555555555example" //Corresponds to NN5 - layer iii -
        connected to instance
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
  ],
}
```

```
{
  "InstanceId": "i-444444444example", //Corresponds to instance 4
  "InstanceType": "trn1.2xlarge",
  "NetworkNodes": [
    "nn-111111111example", //Corresponds to NN1 - layer i
    "nn-333333333example", //Corresponds to NN3 - layer ii
    "nn-666666666example" //Corresponds to NN6 - layer iii -
connected to instance
  ],
  "ZoneId": "usw2-az2",
  "AvailabilityZone": "us-west-2a"
},
"NextToken": "SomeEncryptedToken"
}
```

Batasan

Batasan berikut berlaku:

- Contoh harus di running negara bagian.
- Setiap tampilan topologi instans unik per akun.
- AWS Management Console Tidak mendukung melihat topologi instance.

Prasyarat misalnya topologi

Sebelum Anda menjelaskan topologi instans untuk instans Anda, pastikan instans Anda memenuhi persyaratan berikut.

Persyaratan untuk menggambarkan topologi instans Anda

- [Wilayah AWS](#)
- [Tipe instans](#)
- [Status instans](#)
- [Izin IAM](#)

Wilayah AWS

Didukung Wilayah AWS:

- AS Timur (Virginia Utara), AS Timur (Ohio), AS Barat (California Utara), AS Barat (Oregon)
- Asia Pasifik (Seoul), Asia Pasifik (Tokyo)
- Kanada (Pusat)
- Eropa (Frankfurt), Eropa (Irlandia), Eropa (Stockholm)

Tipe instans

Tipe instans yang didukung:

- hpc6a.48xlarge | hpc6id.32xlarge | hpc7a.12xlarge | hpc7a.24xlarge | hpc7a.48xlarge | hpc7a.96xlarge | hpc7g.4xlarge | hpc7g.8xlarge | hpc7g.16xlarge
- p3dn.24xlarge | p4d.24xlarge | p4de.24xlarge | p5.48xlarge
- trn1.2xlarge | trn1.32xlarge | trn1n.32xlarge

Untuk melihat tipe instans yang tersedia di Wilayah tertentu

Tipe instans yang tersedia berbeda-beda menurut Wilayah. Untuk melihat apakah tipe instans tersedia di Wilayah, gunakan perintah [describe-instance-types-offerings](#) dengan parameter `--region`. Sertakan `--filters` parameter untuk cakupan hasil ke keluarga instans atau tipe instans yang Anda minati dan `--query` parameter untuk cakupan output ke nilai InstanceType.

```
aws ec2 describe-instance-type-offerings \  
  --region us-east-2 \  
  --filters 'Name=instance-type, Values=trn1*' \  
  --query 'InstanceTypeOfferings[].InstanceType'
```

Output yang diharapkan

```
[  
  "trn1.2xlarge",  
  "trn1.32xlarge",  
  "trn1n.32xlarge"  
]
```

Status instans

Instans harus dalam status `running`. Anda tidak bisa mendapatkan informasi topologi instans untuk instans yang berada dalam status lain.

Izin IAM

Identitas IAM Anda (pengguna, grup pengguna, atau peran) memerlukan izin IAM berikut:

- `ec2:DescribeInstanceTopology`

Contoh topologi instans Amazon EC2

Anda dapat menggunakan perintah [describe-instance-topology](#) CLI untuk mendeskripsikan topologi instance untuk instans EC2 Anda.

Saat Anda menggunakan perintah `describe-instance-topology` tanpa parameter atau filter, respons akan menyertakan semua instans yang cocok dengan tipe 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 [Menentukan Wilayah untuk sumber daya](#).

Anda dapat menyertakan parameter untuk mengembalikan instans yang cocok dengan ID instans tertentu atau nama grup penempatan. Anda juga dapat menyertakan filter untuk menampilkan instans yang cocok dengan tipe instans atau keluarga instans tertentu, atau instans di Zona Ketersediaan atau Local Zones tertentu. Anda dapat menyertakan satu parameter atau filter, atau kombinasi parameter dan filter.

Outputnya diberi paginasi, dengan hingga 20 instans per halaman secara default. Anda dapat menentukan hingga 100 instans per halaman menggunakan `--max-results` parameter.

Untuk informasi selengkapnya, lihat [describe-instance-topology](#) di Referensi Perintah AWS CLI .

Izin yang diperlukan

Izin berikut diperlukan untuk menjelaskan topologi instance:

- `ec2:DescribeInstanceTopology`

Contoh

- [Contoh 1 - Tidak ada parameter atau filter](#)
- [Contoh 2 - filter tipe instans](#)
 - [Contoh 2a - Filter pencocokan tepat untuk tipe instans tertentu](#)
 - [Contoh 2b - Filter wild card untuk keluarga instans](#)
 - [Contoh 2c – Gabungan filter keluarga instans dan pencocokan tepat](#)
- [Contoh 3 - filter zona-id](#)
 - [Contoh 3a - Filter Zona Ketersediaan](#)
 - [Contoh 3b - Filter Local Zones](#)
 - [Contoh 3c – Gabungan filter Zona Ketersediaan dan Local Zones](#)
- [Contoh 4 – Gabungan filter tipe instans dan id zona](#)
- [Contoh 5 - Parameter nama grup penempatan](#)
- [Contoh 6 - ID Instans](#)

Contoh 1 - Tidak ada parameter atau filter

Untuk menggambarkan topologi instans dari semua instans Anda

Gunakan perintah CLI [describe-instance-topology](#) tanpa menentukan parameter atau filter apa pun.

```
aws ec2 describe-instance-topology --region us-west-2
```

Respons hanya menampilkan instans yang cocok dengan tipe instans yang didukung untuk API ini. Instans dapat berada di Zona Ketersediaan, Local Zones (ZoneId), dan grup penempatan (GroupName) yang berbeda. Jika instans tidak ada dalam grup penempatan, GroupName kolom tersebut tidak muncul di output. Dalam contoh output berikut, hanya satu instans yang berada dalam grup penempatan.

Contoh output

```
{
  "Instances": [
    {
      "InstanceId": "i-1111111111example",
      "InstanceType": "p4d.24xlarge",
      "GroupName": "my-ml-cpg",
      "NetworkNodes": [
        "nn-1111111111example",
```

```
        "nn-2222222222example",
        "nn-3333333333example"
    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
},
{
    "InstanceId": "i-2222222222example",
    "InstanceType": "p4d.24xlarge",
    "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-3333333333example"
    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
},
{
    "InstanceId": "i-3333333333example",
    "InstanceType": "trn1.32xlarge",
    "NetworkNodes": [
        "nn-1212121212example",
        "nn-1211122211example",
        "nn-1311133311example"
    ],
    "ZoneId": "usw2-az4",
    "AvailabilityZone": "us-west-2d"
},
{
    "InstanceId": "i-4444444444example",
    "InstanceType": "trn1.2xlarge",
    "NetworkNodes": [
        "nn-1111111111example",
        "nn-5434334334example",
        "nn-1235301234example"
    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
}
],
"NextToken": "SomeEncryptedToken"
}
```

Contoh 2 - filter tipe 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 2a - Filter pencocokan tepat untuk tipe instans tertentu

Untuk mendeskripsikan topologi instans dari semua instans Anda yang cocok dengan tipe instans tertentu

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `instance-type`. Dalam contoh ini, output disaring untuk instans `trn1n.32xlarge`. Respons hanya akan mengembalikan instans yang cocok dengan tipe instans yang ditentukan.

```
aws ec2 describe-instance-topology \  
  --region us-west-2 \  
  --filters Name=instance-type,Values=trn1n.32xlarge
```

Contoh output

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "trn1n.32xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    }  
  ],  
  "NextToken": "SomeEncryptedToken"  
}
```

Contoh 2b - Filter wild card untuk keluarga instans

Untuk menggambarkan topologi instans dari semua instans Anda yang cocok dengan keluarga instans

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `instance-type`. Dalam contoh ini, output disaring untuk instans `trn1*`. Respons hanya akan mengembalikan instans yang cocok dengan keluarga instans yang ditentukan.

```
aws ec2 describe-instance-topology \  
  --region us-west-2 \  
  --filters Name=instance-type,Values=trn1*
```

Contoh output

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "trn1n.32xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "usw2-az2",  
      "AvailabilityZone": "us-west-2a"  
    },  
    {  
      "InstanceId": "i-3333333333example",  
      "InstanceType": "trn1.32xlarge",  
      "NetworkNodes": [  
        "nn-1212121212example",  
        "nn-1211122211example",  
        "nn-1311133311example"  
      ],  
      "ZoneId": "usw2-az4",  
      "AvailabilityZone": "us-west-2d"  
    },  
    {  
      "InstanceId": "i-4444444444example",  
      "InstanceType": "trn1.2xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-5434334334example",  
        "nn-1235301234example"  
      ],  
      "ZoneId": "usw2-az2",
```

```

        "AvailabilityZone": "us-west-2a"
    }
],
"NextToken": "SomeEncryptedToken"
}

```

Contoh 2c – Gabungan filter keluarga instans dan pencocokan tepat

Untuk mendeskripsikan topologi instans dari semua instans Anda yang cocok dengan keluarga instans atau tipe instans tertentu

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `instance-type`. Dalam contoh ini, output disaring untuk instans `pd4d*` atau `trn1n.32xlarge`. Respons akan mengembalikan instans yang cocok dengan salah satu filter yang ditentukan.

```

aws ec2 describe-instance-topology \
  --region us-west-2 \
  --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge"

```

Contoh output

```

{
  "Instances": [
    {
      "InstanceId": "i-1111111111example",
      "InstanceType": "p4d.24xlarge",
      "GroupName": "ML-group",
      "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-3333333333example"
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
    {
      "InstanceId": "i-2222222222example",
      "InstanceType": "trn1n.32xlarge",
      "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-4343434343example"
      ]
    }
  ]
}

```

```

    ],
    "ZoneId": "usw2-az2",
    "AvailabilityZone": "us-west-2a"
  }
],
"NextToken": "SomeEncryptedToken"
}

```

Contoh 3 - filter zona-id

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 3a - Filter Zona Ketersediaan

Untuk menjelaskan topologi instans dari semua instans yang cocok dengan Zona Ketersediaan yang ditentukan

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `zone-id`. Dalam contoh ini, output disaring menggunakan ID Availability Zone `use1-az1`. Respons hanya akan menampilkan instans yang cocok dengan Zona Ketersediaan yang ditentukan.

```

aws ec2 describe-instance-topology \
  --region us-east-1 \
  --filters Name=zone-id,Values=use1-az1

```

Contoh output

```

{
  "Instances": [
    {
      "InstanceId": "i-222222222example",
      "InstanceType": "trn1n.32xlarge",
      "NetworkNodes": [
        "nn-111111111example",
        "nn-222222222example",
        "nn-3214313214example"
      ],
      "ZoneId": "use1-az1",
      "AvailabilityZone": "us-east-1a"
    }
  ]
}

```

```

    ],
    "NextToken": "SomeEncryptedToken"
  }

```

Contoh 3b - Filter Local Zones

Untuk menjelaskan topologi instans dari semua instans Anda yang cocok dengan Local Zones yang ditentukan

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `zone-id`. Dalam contoh ini, output disaring menggunakan ID Zona Lokal `use1-atl2-az1`. Respons hanya akan mengembalikan instans yang cocok dengan Local Zones yang ditentukan.

```

aws ec2 describe-instance-topology \
  --region us-east-1 \
  --filters Name=zone-id,Values=use1-atl2-az1

```

Contoh output

```

{
  "Instances": [
    {
      "InstanceId": "i-1111111111example",
      "InstanceType": "p4d.24xlarge",
      "GroupName": "ML-group",
      "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-3333333333example"
      ],
      "ZoneId": "use1-atl2-az1",
      "AvailabilityZone": "us-east-1-atl-2a"
    }
  ],
  "NextToken": "SomeEncryptedToken"
}

```

Contoh 3c – Gabungan filter Zona Ketersediaan dan Local Zones

Untuk mendeskripsikan topologi instans dari semua instans yang cocok dengan Zona Ketersediaan atau Local Zones tertentu

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `zone-id`. Dalam contoh ini, output disaring menggunakan ID Availability Zone `use1-az1` dan Local Zone `use1-atl2-az1`. Respons akan mengembalikan instans yang cocok dengan salah satu filter yang ditentukan.

```
aws ec2 describe-instance-topology \  
  --region us-east-1 \  
  --filters Name=zone-id,Values=use1-az1,use1-atl2-az1
```

Contoh output

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-1111111111example",  
      "InstanceType": "p4d.24xlarge",  
      "GroupName": "ML-group",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "use1-atl2-az1",  
      "AvailabilityZone": "us-east-1-atl-2a"  
    },  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "trn1n.32xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3214313214example"  
      ],  
      "ZoneId": "use1-az1",  
      "AvailabilityZone": "us-east-1a"  
    }  
  ],  
  "NextToken": "SomeEncryptedToken"  
}
```

Contoh 4 – Gabungan filter tipe instans dan id zona

Anda dapat menggabungkan semua filter dalam satu perintah.

Untuk menjelaskan topologi instans dari semua instans yang cocok dengan tipe instans tertentu, keluarga instans, Zona Ketersediaan, atau Local Zones

Gunakan perintah CLI [describe-instance-topology](#) dengan filter `instance-type` dan `zone-id`. Dalam contoh ini, output difilter untuk keluarga `p4d*` instance, tipe instans, `trn1n.32xlarge` ID Zona `use1-az1` Ketersediaan, dan ID Zona `use1-atl2-az1` Lokal. Respons akan mengembalikan instans yang cocok `p4d*` atau instans `trn1n.32xlarge` di zona `us-east-1a` atau `us-east-1-atl-2a`.

```
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"
```

Contoh output

```
{  
  "Instances": [  
    {  
      "InstanceId": "i-1111111111example",  
      "InstanceType": "p4d.24xlarge",  
      "GroupName": "ML-group",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3333333333example"  
      ],  
      "ZoneId": "use1-atl2-az1",  
      "AvailabilityZone": "us-east-1-atl-2a"  
    },  
    {  
      "InstanceId": "i-2222222222example",  
      "InstanceType": "trn1n.32xlarge",  
      "NetworkNodes": [  
        "nn-1111111111example",  
        "nn-2222222222example",  
        "nn-3214313214example"  
      ],  
      "ZoneId": "use1-az1",  
      "AvailabilityZone": "us-east-1a"  
    }  
  ],  
}
```

```
"NextToken": "SomeEncryptedToken"
}
```

Contoh 5 - Parameter nama grup penempatan

Untuk menggambarkan topologi instans dari semua instans Anda dalam grup penempatan tertentu

Gunakan perintah CLI [describe-instance-topology](#) dengan parameter `group-names`. Dalam contoh berikut, instans dapat berada di grup penempatan `ML-group` atau `HPC-group`. Respons akan mengembalikan instans yang ada di salah satu grup penempatan.

```
aws ec2 describe-instance-topology \
  --region us-west-2 \
  --group-names ML-group HPC-group
```

Contoh output

```
{
  "Instances": [
    {
      "InstanceId": "i-111111111example",
      "InstanceType": "p4d.24xlarge",
      "GroupName": "ML-group",
      "NetworkNodes": [
        "nn-111111111example",
        "nn-222222222example",
        "nn-333333333example"
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
    {
      "InstanceId": "i-222222222example",
      "InstanceType": "trn1n.32xlarge",
      "GroupName": "HPC-group",
      "NetworkNodes": [
        "nn-111111111example",
        "nn-222222222example",
        "nn-3214313214example"
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    }
  ]
}
```

```
    }
  ],
  "NextToken": "SomeEncryptedToken"
}
```

Contoh 6 - ID Instans

Untuk menggambarkan topologi instans dari instans tertentu

Gunakan perintah CLI [describe-instance-topology](#) dengan parameter `--instance-ids`. Respons akan mengembalikan instans yang cocok dengan ID instans yang ditentukan.

```
aws ec2 describe-instance-topology \
  --region us-west-2 \
  --instance-ids i-1111111111example i-2222222222example
```

Contoh output

```
{
  "Instances": [
    {
      "InstanceId": "i-1111111111example",
      "InstanceType": "p4d.24xlarge",
      "GroupName": "ML-group",
      "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-3333333333example"
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    },
    {
      "InstanceId": "i-2222222222example",
      "InstanceType": "trn1n.32xlarge",
      "GroupName": "HPC-group",
      "NetworkNodes": [
        "nn-1111111111example",
        "nn-2222222222example",
        "nn-3214313214example"
      ],
      "ZoneId": "usw2-az2",
      "AvailabilityZone": "us-west-2a"
    }
  ]
}
```

```
    }  
  ],  
  "NextToken": "SomeEncryptedToken"  
}
```

Grup penempatan

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 node-to-node komunikasi yang digabungkan secara ketat yang khas dari aplikasi komputasi kinerja tinggi (HPC).
- **Partisi** – menyebarkan instans Anda di seluruh partisi logis sehingga grup instans dalam satu partisi tidak menggunakan bersama perangkat keras yang mendasari dengan grup instans dalam partisi 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.

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.

Tidak ada biaya untuk membuat grup penempatan.

Strategi penempatan

Anda dapat membuat grup penempatan menggunakan salah satu strategi penempatan berikut.

Strategi penempatan:

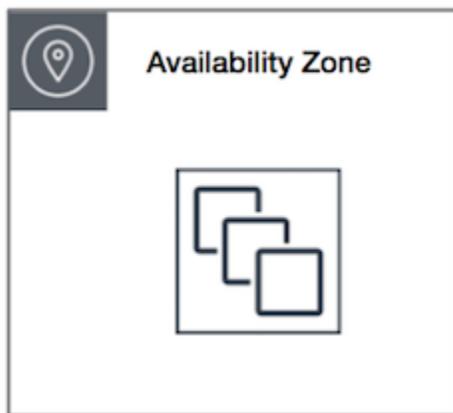
- [Grup penempatan klaster](#)
- [Grup penempatan partisi](#)

- [Grup penempatan tersebar](#)

Grup penempatan kluster

Grup penempatan kluster adalah pengelompokan logis dari instans di dalam Zona Ketersediaan. Grup penempatan kluster dapat menjangkau jaringan privat virtual (VPC) yang di-peering di Wilayah yang sama. Instans dalam grup penempatan kluster yang sama dapat menikmati batas throughput per-aliran yang lebih tinggi untuk lalu lintas TCP/IP dan ditempatkan di segmen bandwidth high-bisection yang sama di jaringan.

Image berikut menunjukkan instans yang ditempatkan dalam grup penempatan kluster.



Grup penempatan kluster 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 memberikan latensi terendah dan kinerja packet-per-second jaringan tertinggi untuk grup penempatan Anda, pilih jenis instans yang mendukung jaringan yang disempurnakan. Untuk informasi lebih lanjut, lihat [Jaringan yang Ditingkatkan](#).

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.

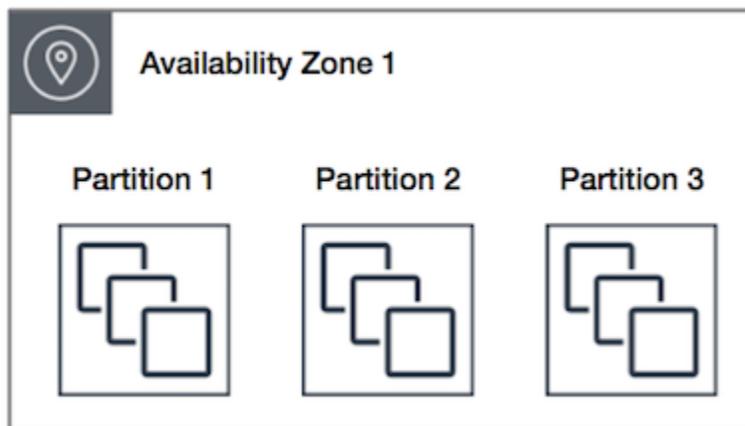
Jika Anda menghentikan satu instans dalam grup penempatan dan kemudian memulainya lagi, ini masih berjalan dalam grup penempatan. Namun demikian, momen mulai gagal jika tidak cukup kapasitas untuk instans.

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 instans dapat memigrasikannya ke perangkat keras yang memiliki kapasitas untuk semua proses yang diminta.

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 dalam partisi – Anda dapat melihat instans mana dan 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 dan 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.

Grup penempatan tersebar

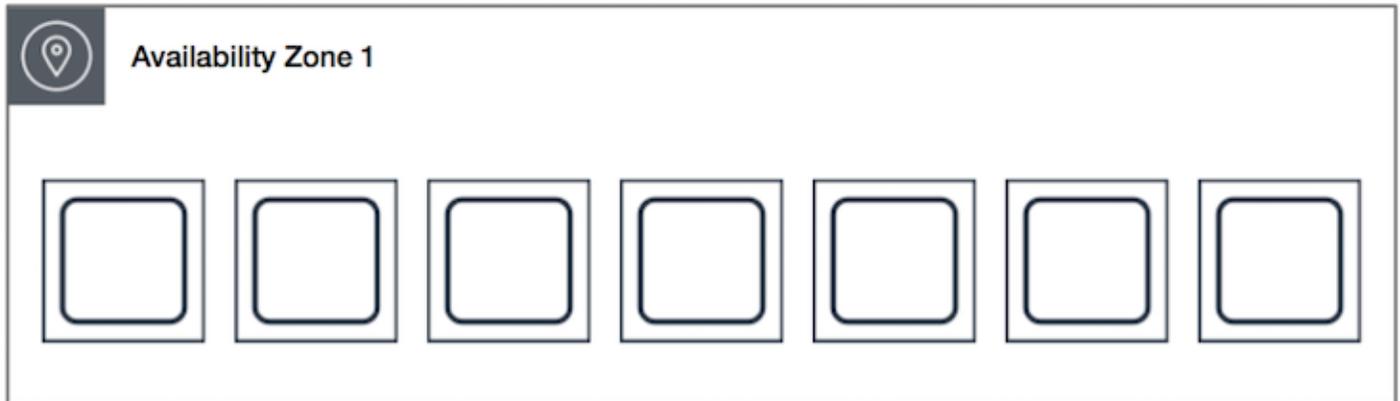
Grup penempatan sebaran adalah kelompok instans yang ditempatkan pada perangkat keras yang berbeda.

Grup penempatan sebaran direkomendasikan untuk aplikasi yang memiliki sejumlah kecil instans penting yang harus disimpan terpisah satu sama lain. Peluncuran instans dalam grup penempatan tingkat sebaran mengurangi risiko kegagalan simultan yang mungkin terjadi ketika instans memiliki peralatan yang sama. Grup penempatan tingkat tersebar menyediakan akses ke perangkat keras yang berbeda, dan oleh karena itu cocok untuk menggabungkan tipe 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 dapat menyebarkan instans di seluruh rak atau host. Grup penempatan spread level rak dapat digunakan di AWS Wilayah dan seterusnya AWS Outposts. Grup penempatan spread level host AWS Outposts hanya dapat digunakan.

Grup penempatan penyebaran tingkat rak

Image berikut menunjukkan tujuh instans dalam satu Zona Ketersediaan yang ditempatkan dalam grup penempatan sebaran. Tujuh instans tersebut ditempatkan di tujuh rak yang berbeda, masing-masing rak memiliki jaringan dan sumber daya sendiri.



Grup penempatan spread level rak dapat menjangkau beberapa Availability Zone di Region yang sama. Di Wilayah, grup penempatan spread level rak dapat memiliki maksimal tujuh instance berjalan per Availability Zone per grup. Dengan Outposts, grup penempatan spread level rak dapat menampung sebanyak mungkin instance karena Anda memiliki rak di penyebaran Outpost Anda.

Grup penempatan sebaran tingkat host

Grup penempatan spread tingkat host hanya tersedia dengan AWS Outposts. Grup penempatan tingkat penyebaran host dapat menampung instance sebanyak yang Anda miliki sebagai host dalam penyebaran Outpost Anda. Untuk informasi selengkapnya, lihat [the section called “Grup penempatan di AWS Outposts”](#).

Aturan dan batasan grup penempatan

Topik

- [Aturan dan pembatasan umum](#)
- [Aturan dan batasan grup penempatan kluster](#)
- [Aturan dan batasan grup penempatan partisi](#)
- [Aturan dan batasan grup penempatan sebaran](#)

Aturan dan pembatasan umum

Sebelum Anda menggunakan grup penempatan, perhatikan aturan berikut ini:

- Anda dapat membuat maksimal 500 grup penempatan per akun di setiap Wilayah.
- Nama yang Anda tentukan untuk grup penempatan harus unik di dalam akun AWS Anda untuk Wilayah tersebut.
- Anda tidak dapat menggabungkan grup penempatan.
- Sebuah instans dapat diluncurkan dalam satu grup penempatan pada satu kesempatan; ini tidak dapat mencakup beberapa grup penempatan.
- [Reservasi Kapasitas Sesuai Permintaan](#) dan [Instans Cadangan zona memungkinkan Anda memesan kapasitas untuk instans EC2 di Availability Zone](#). Saat Anda meluncurkan instance, jika atribut instance cocok dengan yang ditentukan oleh Reservasi Kapasitas Sesuai Permintaan atau Instans Cadangan zona, maka kapasitas cadangan secara otomatis digunakan oleh instans. Ini juga benar jika Anda meluncurkan instance ke dalam grup penempatan.

Jika Anda berencana untuk meluncurkan instance ke dalam grup penempatan kluster, sebaiknya Anda menyimpan kapasitas secara eksplisit di grup penempatan kluster. Anda dapat melakukannya dengan membuat [Reservasi Kapasitas Sesuai Permintaan dalam grup penempatan kluster tertentu](#). Perhatikan bahwa meskipun Anda dapat memesan kapasitas dengan cara ini menggunakan Reservasi Kapasitas Sesuai Permintaan, hal yang sama tidak dapat dilakukan dengan Instans Cadangan zona karena mereka tidak dapat memesan kapasitas secara eksplisit dalam 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.

Aturan dan batasan grup penempatan kluster

Aturan berikut berlaku untuk grup penempatan kluster:

- Berikut ini adalah tipe instans yang didukung:
 - Instans generasi saat ini, kecuali instans [performa burstable](#) (misalnya, T2), instans [Mac1](#), dan [instans M7i-flex](#).
 - Contoh generasi sebelumnya berikut: A1, C3, C4, I2, M4, R3, dan R4.
- Grup penempatan kluster tidak dapat mencakup beberapa Zona Ketersediaan.
- Kecepatan throughput maksimum jaringan di antara dua instans dalam grup penempatan kluster 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.
- Lalu lintas jaringan ke internet dan melalui AWS Direct Connect koneksi ke sumber daya lokal dibatasi hingga 5 Gbps untuk grup penempatan klaster.

Aturan dan batasan grup penempatan partisi

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.

Aturan dan batasan grup penempatan sebaran

Aturan berikut berlaku untuk grup penempatan tersebar:

- Grup penempatan sebaran rak mendukung maksimal tujuh instans yang berjalan per Zona Ketersediaan. Misalnya, di Wilayah dengan tiga Zona Ketersediaan, Anda dapat menjalankan total 21 instans dalam grup, dengan tujuh instans di setiap Zona Ketersediaan. Jika Anda mencoba memulai instans kedelapan dalam Zona Ketersediaan yang sama dan dalam grup penempatan sebaran yang sama, instans tersebut tidak akan diluncurkan. Jika Anda membutuhkan lebih dari

tujuh instans di Zona Ketersediaan, sebaiknya Anda menggunakan beberapa grup penempatan sebaran. Penggunaan beberapa grup penempatan sebaran tidak memberikan jaminan tentang penyebaran instans antar grup, tetapi ini membantu memastikan penyebaran untuk tiap-tiap grup, sehingga membatasi dampak dari kelas kegagalan tertentu.

- Grup penempatan sebaran tidak didukung untuk Instans Khusus.
- Grup penempatan spread tingkat host hanya didukung untuk grup penempatan di AWS Outposts. Grup penempatan spread tingkat host dapat menampung instance sebanyak yang Anda miliki sebagai host dalam penyebaran Outpost Anda.
- Di Wilayah, grup penempatan spread level rak dapat memiliki maksimal tujuh instance berjalan per Availability Zone per grup. Dengan AWS Outposts, grup penempatan spread level rak dapat menampung sebanyak mungkin instance karena Anda memiliki rak di penyebaran Outpost Anda.
- Reservasi Kapasitas tidak menyimpan kapasitas dalam grup penempatan sebaran.

Bekerja dengan grup penempatan

Daftar Isi

- [Buat grup penempatan](#)
- [Lihat informasi grup penempatan](#)
- [Menandai grup penempatan](#)
- [Meluncurkan instans dalam grup penempatan](#)
- [Menjelaskan instans dalam grup penempatan](#)
- [Mengubah grup penempatan untuk instans](#)
- [Menghapus instans dari grup penempatan](#)
- [Menghapus grup penempatan](#)

Buat grup penempatan

Anda dapat membuat salinan dari grup keamanan menggunakan salah satu metode berikut.

Console

Untuk membuat grup penempatan menggunakan konsol

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. Tentukan nama untuk grup tersebut.
5. Pilih strategi penempatan untuk grup tersebut.
 - Jika Anda memilih Sebaran, pilih tingkat sebaran.
 - Rak - tidak ada batasan
 - Host - hanya untuk Outposts
 - Jika Anda memilih Partisi, pilih jumlah partisi di dalam grup.
6. Untuk menandai grup penempatan, pilih Tambahkan tanda, lalu masukkan kunci dan nilai. Pilih Tambahkan tanda untuk setiap tanda yang ingin Anda tambahkan.
7. Pilih Buat grup.

AWS CLI

Untuk membuat grup penempatan menggunakan AWS CLI

Gunakan perintah [create-placement-group](#). Contoh berikut membuat grup penempatan bernama `my-cluster` yang menggunakan strategi penempatan `cluster` serta menerapkan tanda 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 partisi menggunakan AWS CLI

Gunakan perintah [create-placement-group](#). Tentukan parameter `--strategy` dengan nilai `partition`, dan tentukan parameter `--partition-count` dengan jumlah partisi yang diinginkan. Dalam contoh ini, grup penempatan partisi diberi nama `HDFS-Group-A` dan dibuat dengan lima partisi.

```
aws ec2 create-placement-group \  
  --group-name HDFS-Group-A \  
  --strategy partition \  
  --partition-count 5
```

```
--strategy partition \  
--partition-count 5
```

PowerShell

Untuk membuat grup penempatan menggunakan AWS Tools for Windows PowerShell

Gunakan perintah [New-EC2PlacementGroup](#).

Lihat informasi grup penempatan

Anda dapat melihat semua grup penempatan Anda dan informasi tentang mereka menggunakan salah satu metode berikut.

Console

Untuk melihat informasi tentang satu atau beberapa grup penempatan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, di bawah Jaringan & Keamanan, pilih Grup Penempatan.
3. Dalam tabel Grup penempatan, untuk setiap grup penempatan, Anda dapat melihat informasi berikut:
 - Nama grup — Nama yang Anda berikan kepada grup penempatan.
 - Group Id — ID dari grup penempatan.
 - Strategi — Strategi penempatan untuk kelompok penempatan.
 - Negara - Keadaan kelompok penempatan.
 - Partisi — Jumlah partisi. Hanya berlaku jika strateginya adalah partisi.
 - Grup ARN — Nama Sumber Daya Amazon (ARN) dari grup penempatan.

AWS CLI

Untuk mendeskripsikan semua grup penempatan Anda

Gunakan [perintah describe-placement-groups](#) AWS CLI .

```
aws ec2 describe-placement-groups
```

Contoh tanggapan

```
{
  "PlacementGroups": [
    {
      "GroupName": "my-cluster-pg",
      "State": "available",
      "Strategy": "cluster",
      "GroupId": "pg-0123456789example",
      "GroupArn": "arn:aws:ec2:eu-west-1:111111111111:placement-group/my-
cluster-pg"
    },
    ...
  ]
}
```

Untuk menggambarkan grup penempatan tertentu

Gunakan [perintah describe-placement-groups](#) AWS CLI . Anda dapat menentukan parameter `--group-id` atau `--group-name` parameter.

Tentukan ID grup penempatan:

```
aws ec2 describe-placement-groups --group-id pg-0123456789example
```

Tentukan nama grup penempatan:

```
aws ec2 describe-placement-groups --group-name my-cluster-pg
```

Contoh tanggapan

```
{
  "PlacementGroups": [
    {
      "GroupName": "my-cluster-pg",
      "State": "available",
      "Strategy": "cluster",
      "GroupId": "pg-0123456789example",
      "GroupArn": "arn:aws:ec2:eu-west-1:111111111111:placement-group/my-
cluster-pg"
    }
  ]
}
```

```
]
}
```

Menandai grup penempatan

Untuk membantu mengategorikan dan mengelola grup penempatan yang ada, Anda dapat menandai metadata kustom. Untuk informasi lebih lanjut tentang cara kerja tag, lihat [Tandai sumber daya Amazon EC2 Anda](#).

Saat Anda menandai grup penempatan, instans yang diluncurkan ke dalam grup penempatan tidak secara otomatis ditandai. Anda perlu secara eksplisit menandai instans yang diluncurkan ke dalam grup penempatan. Untuk informasi selengkapnya, lihat [Tambahkan tanda saat meluncurkan instans](#).

Anda dapat menampilkan, menambahkan, dan menghapus tanda menggunakan salah satu metode berikut.

Console

Untuk melihat, menambahkan, atau menghapus tanda untuk grup penempatan yang ada

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Grup Penempatan.
3. Pilih grup penempatan, lalu pilih Tindakan, Kelola tanda.
4. Layar Kelola tanda menampilkan tanda yang ditetapkan ke grup penempatan ini.
 - Untuk menambahkan tag, pilih Tambahkan tag, dan masukkan kunci dan nilai tag. Anda dapat menambahkan hingga 50 tanda per grup penempatan. Untuk informasi selengkapnya, lihat [Pembatasan tanda](#).
 - Untuk menghapus tag, pilih Hapus di samping tanda yang ingin Anda hapus.
5. Pilih Simpan.

AWS CLI

Untuk melihat tanda grup penempatan

Gunakan perintah [describe-tags](#) untuk melihat tanda sumber daya yang ditentukan. Dalam contoh berikut, Anda mendeskripsikan tanda untuk semua kunci publik Anda.

```
aws ec2 describe-tags \  
  --filters Name=resource-type,Values=placement-group
```

```
{  
  "Tags": [  
    {  
      "Key": "Environment",  
      "ResourceId": "pg-0123456789EXAMPLE",  
      "ResourceType": "placement-group",  
      "Value": "Production"  
    },  
    {  
      "Key": "Environment",  
      "ResourceId": "pg-9876543210EXAMPLE",  
      "ResourceType": "placement-group",  
      "Value": "Production"  
    }  
  ]  
}
```

Anda juga dapat menggunakan perintah [describe-tags](#) guna melihat tanda untuk grup penempatan dengan menentukan ID-nya. Dalam contoh berikut, Anda menjelaskan tanda untuk `pg-0123456789EXAMPLE`.

```
aws ec2 describe-tags \  
  --filters Name=resource-id,Values=pg-0123456789EXAMPLE
```

```
{  
  "Tags": [  
    {  
      "Key": "Environment",  
      "ResourceId": "pg-0123456789EXAMPLE",  
      "ResourceType": "placement-group",  
      "Value": "Production"  
    }  
  ]  
}
```

Anda juga dapat melihat tanda grup penempatan dengan mendeskripsikan grup penempatan.

Gunakan perintah [describe-placement-groups](#) untuk melihat konfigurasi grup penempatan yang ditentukan, yang mencakup setiap tanda yang ditentukan untuk grup penempatan.

```
aws ec2 describe-placement-groups \  
  --group-name my-cluster
```

```
{  
  "PlacementGroups": [  
    {  
      "GroupName": "my-cluster",  
      "State": "available",  
      "Strategy": "cluster",  
      "GroupId": "pg-0123456789EXAMPLE",  
      "Tags": [  
        {  
          "Key": "Environment",  
          "Value": "Production"  
        }  
      ]  
    }  
  ]  
}
```

Untuk menandai grup penempatan yang ada menggunakan AWS CLI

Anda dapat menggunakan perintah [create-tags](#) untuk menandai sumber daya yang ada. Dalam contoh berikut, grup penempatan yang ada diberi tanda dengan Key=Cost-Center dan Value=CC-123.

```
aws ec2 create-tags \  
  --resources pg-0123456789EXAMPLE \  
  --tags Key=Cost-Center,Value=CC-123
```

Untuk menghapus tag dari grup penempatan menggunakan AWS CLI

Anda dapat menggunakan perintah [delete-tags](#) untuk menghapus tanda dari sumber daya yang ada. Sebagai contoh, lihat [Contoh](#) dalam Referensi Perintah AWS CLI .

PowerShell

Untuk melihat tanda grup penempatan

Gunakan perintah [Get-EC2Tag](#).

Untuk menjelaskan tanda grup penempatan tertentu

Gunakan perintah [Get-EC2PlacementGroup](#).

Untuk menandai grup penempatan yang sudah ada

Gunakan perintah [New-EC2Tag](#).

Untuk menghapus tanda dari grup penempatan

Gunakan perintah [Remove-EC2Tag](#).

Meluncurkan instans dalam grup penempatan

Anda dapat meluncurkan instans ke dalam grup penempatan jika [aturan dan pembatasan grup penempatan terpenuhi](#) menggunakan salah satu metode berikut.

Console

Untuk meluncurkan instans ke grup penempatan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari dasbor konsol EC2, di kotak Luncurkan instans, pilih Luncurkan instans. Lengkapi formulir sesuai petunjuk, berhati-hatilah saat melakukan hal berikut:
 - Di bawah Tipe instans, pilih tipe instans yang dapat diluncurkan ke dalam grup penempatan.
 - Di kotak Ringkasan, di bawah Jumlah instans, masukkan jumlah instans yang Anda butuhkan di grup penempatan ini, karena Anda mungkin tidak dapat menambahkan instans ke grup penempatan nanti.
 - Di bawah Detail lanjutan, untuk Nama grup penempatan, Anda dapat memilih untuk menambahkan instans ke grup penempatan baru atau yang sudah ada. Jika Anda memilih grup penempatan dengan strategi partisi, untuk partisi Target, pilih partisi untuk meluncurkan instans.

AWS CLI

Untuk meluncurkan instans ke grup penempatan

Gunakan perintah [run-instances](#) dan tentukan nama grup penempatan menggunakan parameter `--placement "GroupName = my-cluster"`. Dalam contoh ini, grup penempatan diberi nama `my-cluster`.

```
aws ec2 run-instances --placement "GroupName = my-cluster"
```

Untuk meluncurkan instance ke partisi tertentu dari grup penempatan partisi menggunakan AWS CLI

Gunakan perintah [run-instances](#) dan tentukan nama grup penempatan dan partisi menggunakan parameter `--placement "GroupName = HDFS-Group-A, PartitionNumber = 3"`. Dalam contoh ini, grup penempatan diberi nama `HDFS-Group-A` dan nomor partisinya adalah 3.

```
aws ec2 run-instances --placement "GroupName = HDFS-Group-A, PartitionNumber = 3"
```

PowerShell

Untuk meluncurkan instans ke grup penempatan menggunakan AWS Tools for Windows PowerShell

Gunakan [New-EC2Instance](#) perintah dan tentukan nama grup penempatan menggunakan `-Placement_GroupName` parameter.

Menjelaskan instans dalam grup penempatan

Anda dapat melihat informasi penempatan instans Anda menggunakan salah satu metode berikut. Anda juga dapat membuat filter grup penempatan partisi sesuai nomor partisi menggunakan AWS CLI.

Console

Untuk melihat grup penempatan dan nomor partisi suatu instans

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 Grup Host dan penempatan, temukan grup Penempatan. Jika instans tersebut tidak ada dalam grup penempatan, kolomnya akan kosong. Atau, mungkin

berisi nama dari grup penempatan. Jika grup penempatannya adalah grup penempatan partisi, Nomor partisi berisi nomor partisi untuk instans tersebut.

AWS CLI

Untuk melihat nomor partisi suatu instans di grup penempatan partisi

Gunakan perintah [describe-instances](#) dan tentukan parameter `--instance-id`.

```
aws ec2 describe-instances --instance-id i-0123a456700123456
```

Jawaban tersebut berisi informasi penempatan, yang mencakup nama grup penempatan dan nomor partisi untuk instans tersebut.

```
"Placement": {
  "AvailabilityZone": "us-east-1c",
  "GroupName": "HDFS-Group-A",
  "PartitionNumber": 3,
  "Tenancy": "default"
}
```

Untuk memfilter instans untuk grup penempatan partisi dan nomor partisi tertentu

Gunakan perintah [describe-instances](#) dan tentukan parameter `--filters` dengan filter `placement-group-name` dan `placement-partition-number`. Dalam contoh ini, grup penempatan diberi nama `HDFS-Group-A` dan nomor partisinya adalah `7`.

```
aws ec2 describe-instances --filters "Name = placement-group-name, Values = HDFS-Group-A" "Name = placement-partition-number, Values = 7"
```

Responsnya mencakup semua instans yang ada dalam partisi tertentu di grup penempatan tertentu. Berikut adalah contoh output yang hanya menunjukkan ID instans, tipe instans, dan informasi penempatan untuk instans yang dikembalikan.

```
"Instances": [
  {
    "InstanceId": "i-0a1bc23d4567e8f90",
    "InstanceType": "r4.large",
  },
  "Placement": {
```

```
        "AvailabilityZone": "us-east-1c",
        "GroupName": "HDFS-Group-A",
        "PartitionNumber": 7,
        "Tenancy": "default"
    }
  {
    "InstanceId": "i-0a9b876cd5d4ef321",
    "InstanceType": "r4.large",
  },
  "Placement": {
    "AvailabilityZone": "us-east-1c",
    "GroupName": "HDFS-Group-A",
    "PartitionNumber": 7,
    "Tenancy": "default"
  }
],
```

Mengubah grup penempatan untuk instans

Anda dapat mengubah grup penempatan suatu instans dengan cara berikut:

- Pindahkan suatu instans yang ada ke grup penempatan
- Pindahkan satu instans dari satu grup penempatan ke grup penempatan lainnya

Sebelum dapat dipindahkan, instans harus ada dalam status stopped.

Console

Untuk memindahkan instans ke grup penempatan

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.
4. Dengan instans yang dipilih, pilih Tindakan, Pengaturan instans, Modifikasi penempatan instans.
5. Untuk Grup penempatan, pilih grup penempatan yang akan menjadi tujuan pemindahan instans.

6. Pilih Simpan.

AWS CLI

Untuk memindahkan instans ke grup penempatan

1. Hentikan instans menggunakan perintah [stop-instances](#).
2. Gunakan perintah [modify-instance-placement](#) dan tentukan nama grup penempatan yang akan menjadi tujuan pemindahan instans.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name MySpreadGroup
```

3. Mulai instans dengan menggunakan perintah [start-instances](#).

PowerShell

Untuk memindahkan instans ke grup penempatan menggunakan AWS Tools for Windows PowerShell

1. Hentikan instance menggunakan [Stop-EC2Instance](#) perintah.
2. Gunakan [Edit-EC2InstancePlacement](#) perintah dan tentukan nama grup penempatan untuk memindahkan instance.
3. Mulai instance menggunakan [Start-EC2Instance](#) perintah.

Menghapus instans dari grup penempatan

Anda menghapus instans dari grup penempatan menggunakan salah satu metode berikut.

Sebelum dapat dihapus dari grup penempatan, instans harus ada dalam status `stopped`.

Console

Untuk menghapus instans dari grup penempatan

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.
4. Dengan instans yang dipilih, pilih Tindakan, Pengaturan instans, Modifikasi penempatan instans.
5. Untuk Grup penempatan, pilih Tidak ada.
6. Pilih Simpan.

AWS CLI

Untuk menghapus instans dari grup penempatan

1. Hentikan instans menggunakan perintah [stop-instances](#).
2. Gunakan perintah [modify-instances-placement](#) dan tentukan string kosong untuk nama grup penempatan.

```
aws ec2 modify-instance-placement \  
  --instance-id i-0123a456700123456 \  
  --group-name ""
```

3. Mulai instans dengan menggunakan perintah [start-instances](#).

PowerShell

Untuk menghapus instans dari grup penempatan menggunakan AWS Tools for Windows PowerShell

1. Hentikan instance menggunakan [Stop-EC2Instance](#) perintah.
2. Gunakan [Edit-EC2InstancePlacement](#) perintah dan tentukan string kosong untuk nama grup penempatan.
3. Mulai instance menggunakan [Start-EC2Instance](#) perintah.

Menghapus grup penempatan

Jika Anda perlu mengganti grup penempatan atau tidak lagi memerlukannya, Anda dapat menghapusnya. Anda dapat menghapus grup penempatan menggunakan salah satu metode berikut.

Prasyarat

Sebelum Anda dapat menghapus grup penempatan, grup penempatan harus tidak berisi instans. Anda dapat [mengakhiri](#) semua instans yang diluncurkan di grup penempatan, [memindahkan](#) instans ke grup penempatan lain, atau [menghapus](#) instans dari grup penempatan.

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-placement-group](#) dan tentukan nama grup penempatan untuk menghapus grup penempatan. Dalam contoh ini, nama grup penempatannya adalah `my-cluster`.

```
aws ec2 delete-placement-group --group-name my-cluster
```

PowerShell

Untuk menghapus grup penempatan menggunakan AWS Tools for Windows PowerShell

Gunakan [Remove-EC2PlacementGroup](#) perintah untuk menghapus grup penempatan.

Membagikan grup penempatan

Berbagi grup penempatan memungkinkan Anda memengaruhi penempatan instance yang saling bergantung yang dimiliki oleh akun terpisah AWS . Anda dapat berbagi grup penempatan di beberapa AWS akun atau di dalam organisasi Anda. Anda dapat meluncurkan instans dalam grup penempatan bersama.

Pemilik grup penempatan dapat berbagi grup penempatan dengan:

- AWS Akun spesifik di dalam atau di luar organisasinya

- Unit organisasi di dalam organisasi -nya
- Seluruh organisasi -nya

Note

AWS Akun tempat Anda ingin berbagi grup penempatan harus memiliki izin berikut dalam kebijakan IAM.

- `ec2:PutResourcePolicy`
- `ec2>DeleteResourcePolicy`

Topik

- [Aturan dan batasan](#)
- [Berbagi di seluruh Zona Ketersediaan](#)
- [Membagikan grup penempatan](#)
- [Identifikasi grup penempatan bersama](#)
- [Luncurkan instans dalam grup penempatan bersama](#)
- [Membatalkan pembagian grup penempatan bersama](#)

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 AWS akun Anda. 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 Anda atau unit organisasi di organisasi Anda, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi lebih lanjut, lihat [Berbagi sumber daya AWS Anda](#).

- Anda bertanggung jawab untuk mengelola instans yang dimiliki oleh Anda dalam grup penempatan bersama.
- Anda tidak dapat melihat atau memodifikasi instans dan reservasi kapasitas yang terkait dengan grup penempatan bersama tetapi tidak dimiliki oleh Anda.

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, Availability Zone us-east-1a untuk AWS akun Anda mungkin tidak memiliki lokasi yang sama dengan AWS akun lain. us-east-1a

Untuk mengidentifikasi lokasi Host Khusus Anda yang terkait dengan akun Anda, Anda harus menggunakan ID Zona Ketersediaan (AZ ID). ID Zona Ketersediaan 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 melihat ID Zona Ketersediaan untuk Zona Ketersediaan di akun Anda

1. Buka AWS RAM konsol di <https://console.aws.amazon.com/ram>.
2. ID Zona Ketersediaan untuk Wilayah saat ini ditampilkan di bawah ID AZ Anda di panel kanan.

Membagikan grup 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 di seluruh AWS akun. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan berbagi dengan mereka.

Jika Anda adalah bagian dari organisasi dalam AWS Organizations berbagi dalam organisasi Anda diaktifkan, konsumen di organisasi Anda diberikan akses ke grup penempatan bersama.

Jika grup penempatan dibagikan dengan AWS akun di luar organisasi Anda, pemilik AWS akun akan menerima undangan untuk bergabung dengan pembagian sumber daya. Mereka dapat mengakses grup penempatan bersama setelah menerima undangan.

Anda dapat berbagi grup penempatan di seluruh AWS akun menggunakan <https://console.aws.amazon.com/ram> atau AWS CLI.

AWS RAM console

Untuk berbagi grup penempatan yang Anda miliki menggunakan <https://console.aws.amazon.com/ram>, lihat [Membuat berbagi sumber daya](#).

AWS CLI

Untuk berbagi grup penempatan yang Anda miliki, gunakan perintah [create-resource-share](#).

Identifikasi grup penempatan bersama

Nama Sumber Daya Amazon (ARN) dari grup penempatan berisi 12 digit ID akun akun yang memiliki grup penempatan. Anda dapat menggunakan ID akun untuk mengidentifikasi pemilik grup penempatan yang dibagikan dengan Anda.

Anda dapat menemukan grup penempatan ARN menggunakan salah satu metode berikut. Untuk informasi selengkapnya, lihat [Lihat informasi grup penempatan](#).

Amazon EC2 console

Untuk mengidentifikasi grup penempatan bersama

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, di bawah Jaringan & Keamanan, pilih Grup Penempatan.
3. Tabel Grup penempatan mencantumkan semua grup penempatan yang dimiliki oleh Anda dan dibagikan dengan Anda. Kolom Grup ARN menampilkan grup penempatan ARN.

Jika kolom ARN Grup tidak terlihat, pilih pengaturan



di sudut kanan atas, aktifkan ARN Grup, dan pilih Konfirmasi.

AWS CLI

Untuk mengidentifikasi grup penempatan bersama

Gunakan [describe-placement-groups](#) perintah untuk membuat daftar semua grup penempatan yang dimiliki oleh Anda dan dibagikan dengan Anda. Sebagai tanggapan, GroupId parameter menampilkan ARN dari grup penempatan.

Luncurkan instans dalam grup penempatan bersama

Important

Saat menggunakan AWS CLI untuk meluncurkan instance dalam grup penempatan bersama, Anda harus menentukan ID grup penempatan dengan menggunakan GroupId parameter.

Anda dapat menggunakan nama grup penempatan hanya jika Anda adalah pemilik grup penempatan yang dibagikan. Sebaiknya gunakan ID grup penempatan untuk menghindari potensi tabrakan nama grup penempatan antar AWS akun.

Anda dapat menemukan ID grup penempatan di konsol Amazon EC2 di Grup Penempatan layar atau dengan menggunakan perintah. [describe-placement-groups](#) AWS CLI Untuk informasi selengkapnya, lihat [Lihat informasi grup penempatan](#).

Console

Untuk meluncurkan instance ke grup penempatan bersama

1. Ikuti prosedur untuk [meluncurkan instance](#), tetapi jangan meluncurkan instance sampai Anda menyelesaikan langkah-langkah berikut untuk menentukan pengaturan untuk grup penempatan.
2. Pada Tipe instans, pilih tipe instans yang didukung. Untuk informasi selengkapnya, lihat [Aturan dan batasan grup penempatan](#).
3. Perluas Detail lanjutan, dan konfigurasi pengaturan grup penempatan sebagai berikut:
 - a. Untuk grup Penempatan, pilih grup penempatan yang dibagikan dengan Anda.

Note

Jika ada grup penempatan dengan nama yang sama, periksa ID grup penempatan untuk memastikan bahwa Anda memilih grup penempatan yang benar.

- b. Jika Anda memilih grup penempatan dengan strategi partisi, untuk partisi Target, pilih partisi untuk meluncurkan instance.
4. Di panel Ringkasan, lakukan hal berikut:

- a. Untuk Jumlah instans, masukkan jumlah instans yang Anda butuhkan dalam grup penempatan ini, karena Anda mungkin tidak dapat menambahkan instans ke grup penempatan nanti.
- b. Tinjau konfigurasi instans Anda, lalu pilih Launch instance.

Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

AWS CLI

Untuk meluncurkan instans dalam grup penempatan bersama

Gunakan [run-instances](#) perintah dan tentukan ID grup penempatan dari grup penempatan bersama.

```
aws ec2 run-instances --placement "GroupId = pg-0123456789example"
```

Untuk meluncurkan instans ke partisi tertentu dari grup penempatan partisi bersama

Gunakan [run-instances](#) perintah dan tentukan ID grup penempatan dan nomor partisi grup penempatan bersama.

```
aws ec2 run-instances --placement "GroupId = pg-0123456789example, PartitionNumber = 3"
```

Tip

Gunakan peering VPC untuk menghubungkan instans yang dimiliki oleh AWS akun terpisah dan dapatkan manfaat latensi penuh yang ditawarkan oleh grup penempatan cluster bersama. Untuk informasi selengkapnya, lihat [Apa yang itu peering VPC?](#)

Membatalkan pembagian grup penempatan bersama

Pemilik grup penempatan dapat membatalkan pembagian grup penempatan bersama kapan saja.

Saat Anda membatalkan berbagi grup penempatan bersama, perubahan berikut ini akan berlaku.

- AWS Akun tempat grup penempatan dibagikan tidak akan lagi dapat meluncurkan instance atau kapasitas cadangan.
- Jika instans Anda berjalan di grup penempatan bersama, instans tersebut akan dipisahkan dari grup penempatan tetapi terus berjalan normal di akun AWS Anda.
- Jika Anda memiliki reservasi kapasitas dalam grup penempatan bersama, mereka akan dipisahkan dari grup penempatan tetapi Anda akan terus memiliki akses ke mereka di akun Anda AWS .

Anda dapat membatalkan pembagian grup penempatan bersama menggunakan salah satu metode berikut.

AWS RAM console

Untuk membatalkan berbagi grup penempatan bersama menggunakan <https://console.aws.amazon.com/ram>, lihat [Menghapus berbagi sumber daya](#).

AWS CLI

Untuk membatalkan berbagi grup penempatan bersama menggunakan AWS Command Line Interface, gunakan perintah [disassociate-resource-share](#).

Grup penempatan 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 AWS Wilayah, 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 suatu AWS Wilayah.

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 instans di seluruh host memungkinkan Anda menggunakan strategi penyebaran dengan satu rak Outpost.

Pertimbangan-pertimbangan

- Grup penempatan spread level rak dapat menampung sebanyak mungkin instance karena Anda memiliki rak di penyebaran Outpost Anda.
- Grup penempatan spread tingkat host dapat menampung instance sebanyak yang Anda miliki sebagai host dalam penyebaran 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 penyebaran, Anda dapat memilih spread tingkat host atau rak untuk menentukan bagaimana grup akan 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.

Aturan berikut berlaku untuk instans yang berada dalam Wavelength Zone:

- Lalu lintas yang keluar dari satu instans ke instans lain dalam VPC di Wavelength Zone yang sama memiliki MTU 1300.
- Lalu lintas yang keluar dari satu instans ke instans lain yang menggunakan IP operator dalam Wavelength Zone memiliki MTU 1500.
- Lalu lintas yang keluar dari satu instans ke instans lain di antara Wavelength Zone dan Wilayah yang menggunakan alamat IP publik memiliki MTU 1500.
- Lalu lintas yang keluar dari satu instans ke instans lain di antara Wavelength Zone dan Wilayah yang menggunakan alamat IP privat memiliki MTU 1300.

Aturan berikut berlaku untuk instans yang berada di Outposts:

- Lalu lintas yang bergerak dari instans di Outposts ke instans di Wilayah memiliki MTU sebesar 1300.

Daftar Isi

- [Frame jumbo \(9001 MTU\)](#)
- [Path MTU Discovery](#)
- [Periksa MTU jalur di antara dua host](#)
- [Periksa MTU untuk contoh Anda](#)
- [Tetapkan MTU untuk instans Anda](#)
- [Pemecahan Masalah](#)

Frame jumbo (9001 MTU)

Frame jumbo memungkinkan lebih dari 1500 byte data dengan meningkatkan ukuran payload per paket, dan dengan demikian 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 melalui gateway internet
- Lalu lintas melalui koneksi peering VPC antar wilayah
- Lalu lintas melalui koneksi VPN
- Lalu lintas di luar AWS wilayah tertentu

Jika paket lebih dari 1500 byte, paket tersebut akan difragmentasi, atau paket-paket tersebut akan diturunkan jika flag Don't Fragment 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 semaksimal mungkin, dan dianjurkan dalam kasus ini. Untuk informasi selengkapnya, lihat [Grup penempatan](#).

Anda dapat menggunakan frame jumbo untuk lalu lintas antara VPC dan jaringan on-premise Anda melalui AWS Direct Connect. Untuk informasi lebih lanjut, dan cara memverifikasi kemampuan Jumbo Frame, lihat [Mengatur MTU Jaringan](#) di Panduan Pengguna AWS Direct Connect .

Semua jenis instans Amazon EC2 mendukung 1500 MTU dan semua jenis instans generasi saat ini mendukung bingkai jumbo. Jenis instans generasi sebelumnya berikut mendukung bingkai jumbo: A1, C3, I2, M3, dan R3.

Untuk informasi selengkapnya tentang ukuran MTU yang didukung, lihat:

- Untuk gateway NAT, lihat [Dasar-dasar gateway NAT](#) di Panduan Pengguna Amazon VPC.
- Untuk gateway transit, lihat [MTU](#) di Panduan Pengguna Gateway Transit Amazon VPC.
- Untuk Local Zones, lihat [Pertimbangan](#) di Panduan Pengguna AWS Local Zones.

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 mentransmisikannya 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 mentransmisikannya 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 atau jika aturan grup keamanan Anda mengizinkan lalu lintas ICMP masuk, Anda dapat menerima respons 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.

Periksa MTU jalur di antara dua host

Anda dapat memeriksa jalur MTU antara instans EC2 Anda dan host lain. Anda dapat menentukan nama DNS atau alamat IP sebagai tujuan. Jika tujuannya adalah instans EC2 lain, verifikasi bahwa grup keamanannya memungkinkan lalu lintas UDP masuk.

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

Instans Linux

Jalankan tracepath perintah pada instans Anda untuk memeriksa jalur MTU antara instans 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 instans 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 <http://www.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 instans EC2 Anda dan tujuan yang ditentukan. Contoh ini memeriksa jalur MTU antara instans 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 instans EC2 Anda. Jika antarmuka jaringan utama tidak `eth0`, ganti `eth0` dengan antarmuka jaringan Anda.

```
[ec2-user ~]$ ip link show eth0
```

Dalam contoh keluaran 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 instans EC2 Anda. Gunakan wildcard (tanda bintang) 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 instans EC2 Anda. Periksa output untuk nama antarmuka `MTU`. 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 instans 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 instans 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
```

Tetapkan MTU untuk instans 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 ke 1500, tetapi Anda bisa menggunakan 9001 sebagai gantinya.

```
[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.
 - Untuk Amazon Linux 2, tambahkan baris berikut ke file `/etc/sysconfig/network-scripts/ifcfg-eth0`:

```
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;
```

- Untuk Amazon Linux AMI, tambahkan baris berikut ke `/etc/dhcp/dhclient-eth0.conf` file Anda.

```
interface "eth0" {  
supersede interface-mtu 1500;  
}
```

- Untuk distribusi Linux lainnya, lihat dokumentasi spesifiknya.
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 instans 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 instans 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 klaster Amazon Redshift saat menggunakan bingkai jumbo, [lihat Kueri Muncul untuk Digantung di Panduan Manajemen Pergeseran Merah Amazon](#).

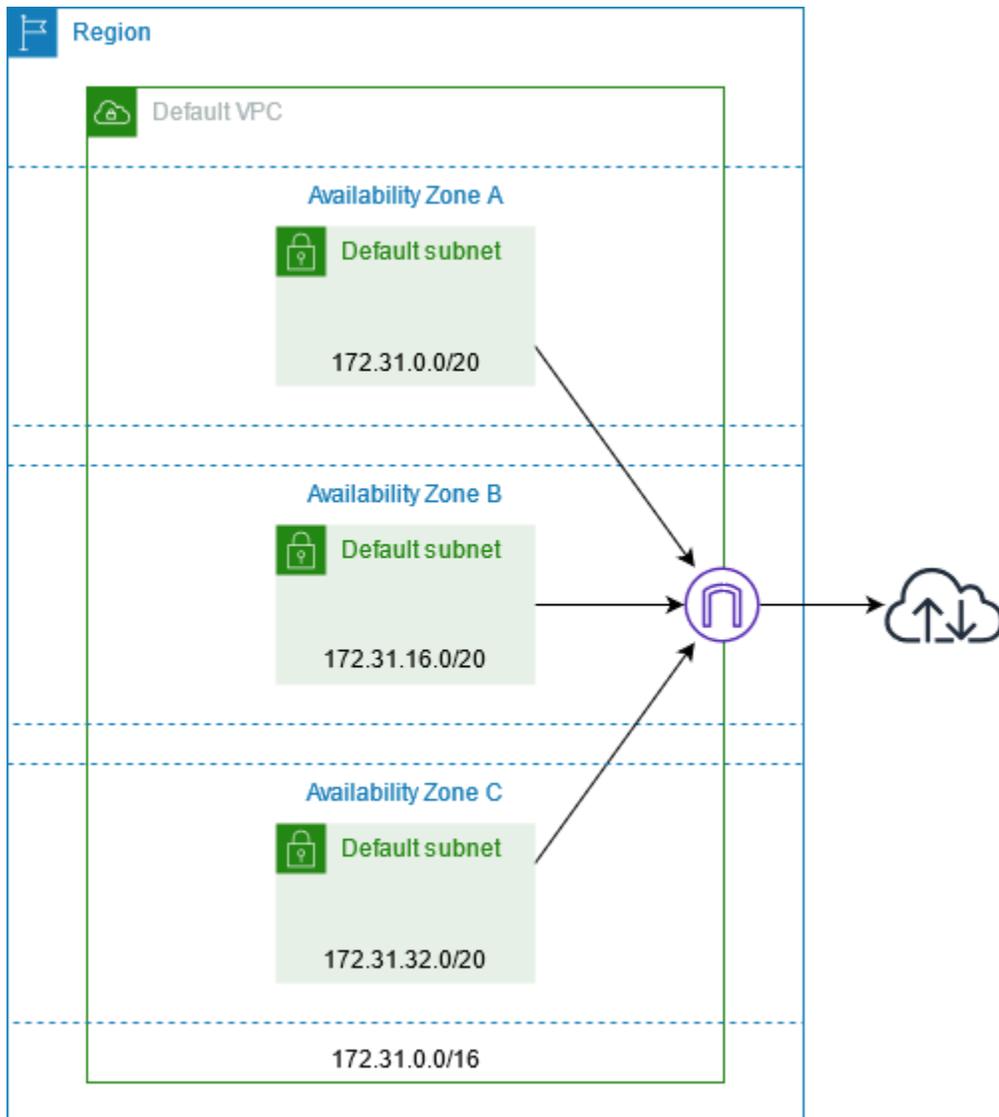
Virtual private cloud untuk instans EC2 Anda

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.

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 Zona Ketersediaan 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. Atau, Anda dapat membuat VPC Anda sendiri dan mengonfigurasinya untuk memenuhi kebutuhan Anda.



Membuat VPC tambahan

Gunakan prosedur berikut untuk membuat VPC dengan subnet, gateway, dan konfigurasi perutean yang Anda butuhkan.

Untuk membuat VPC

1. Buka konsol Amazon VPC di <https://console.aws.amazon.com/vpc/>.
2. Pilih Buat VPC.
3. Pada Sumber daya yang akan dibuat, pilih VPC dan lainnya.

4. Untuk Pembuatan otomatis tanda nama, masukkan nama untuk VPC.
5. Untuk blok IPv4 CIDR, simpan saran default, masukkan blok CIDR yang diperlukan oleh aplikasi atau jaringan Anda.
6. Untuk Jumlah Zona Ketersediaan, pilih 2, sehingga Anda dapat meluncurkan instans di beberapa Zona Ketersediaan untuk memastikan ketersediaan tinggi.
7. Jika instans Anda harus dapat diakses dari internet, lakukan salah satu hal berikut:
 - Jika instans Anda dapat berada di subnet publik, pilih nilai bukan nol untuk Jumlah subnet publik. Simpan kedua opsi di bawah opsi DNS yang dipilih. Anda dapat menambahkan subnet privat secara opsional sekarang atau nanti.
 - Jika instans Anda harus berada dalam subnet privat, pilih 0 untuk Jumlah subnet publik. Untuk Jumlah subnet privat, pilih nomor tergantung pada kebutuhan Anda (nilai yang mungkin sesuai dengan 1 atau 2 subnet privat per Zona Ketersediaan). Untuk gateway NAT, jika instans Anda di kedua Zona Ketersediaan mengirim atau menerima volume lalu lintas yang signifikan di seluruh Zona Ketersediaan, pilih 1 per AZ. Jika tidak, pilih Dalam 1 AZ dan luncurkan instans yang mengirim atau menerima lalu lintas zona di Zona Ketersediaan yang sama dengan gateway NAT.
8. Perluas Kustomisasi blok CIDR subnet. Simpan saran default, atau masukkan blok CIDR untuk setiap subnet. Untuk informasi selengkapnya, lihat [Blok CIDR Subnet](#) di Panduan Pengguna Amazon VPC.
9. Tinjau panel Pratinjau, yang menampilkan sumber daya VPC yang akan dibuat berdasarkan pilihan Anda.
10. Pilih Buat VPC.

Mengakses internet dari instans Anda

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 grup keamanan atau antarmuka jaringan apa pun yang mereka tentukan.
- Peserta dapat memulai, menghentikan, mengakhiri, 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 mendeskripsikan 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 selengkapnya, lihat, [Bagikan VPC Anda dengan akun lain](#) di Panduan Pengguna Amazon VPC.

Subnet khusus IPv6

Instans EC2 yang diluncurkan di subnet khusus IPv6 menerima alamat IPv6, tetapi tidak alamat IPv4. Setiap instance yang Anda luncurkan ke subnet khusus IPv6 harus berupa [instance](#) yang dibangun di Sistem Nitro. AWS

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 antara Anda AWS dan 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. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari [Program AWS Kepatuhan Program AWS Kepatuhan](#) . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon EC2, lihat [AWS Layanan dalam Lingkup berdasarkan AWS Layanan Program Kepatuhan dalam Lingkup oleh Program](#) .
- 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 Windows 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](#)
- [Mengakses Amazon EC2 menggunakan titik akhir VPC antarmuka](#)
- [Manajemen pembaruan untuk instans Windows 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](#)
- [Credential Guard untuk instance Windows](#)

Perlindungan data dalam Amazon EC2

[Model tanggung jawab AWS bersama model tanggung](#) 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. Lihat informasi yang lebih lengkap tentang privasi data dalam [Pertanyaan Umum Privasi Data](#). Lihat informasi tentang perlindungan data di Eropa di pos blog [Model Tanggung Jawab Bersama dan GDPR AWS](#) di Blog Keamanan AWS .

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 sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.

- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-2 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi yang lebih lengkap tentang titik akhir FIPS yang tersedia di [Standar Pemrosesan Informasi Federal \(FIPS\) 140-2](#).

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 dalam transit](#)

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 cipher XTS-AES-256 yang diimplementasikan pada modul perangkat keras di instans tersebut. 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 XTS-AES-256 dan kunci sekali pakai.

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:

- Contoh dengan prosesor AWS Graviton. AWS Graviton2, AWS Graviton3, dan Graviton3E mendukung enkripsi memori yang selalu aktif AWS. 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). Instans dengan prosesor

AMD EPYC generasi ke-3 (Milan) juga mendukung AMD Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP).

Enkripsi dalam transit

Enkripsi pada lapisan fisik

Semua data yang mengalir di seluruh AWS Wilayah melalui jaringan AWS global secara otomatis dienkripsi pada lapisan fisik sebelum meninggalkan fasilitas yang AWS aman. Semua lalu lintas antara AZ akan dienkripsi. Lapisan-lapisan enkripsi tambahan, termasuk yang tercantum dalam bagian ini, dapat memberikan perlindungan tambahan.

Enkripsi disediakan oleh peering VPC Amazon dan Peering lintas wilayah Gateway Transit

Semua lalu lintas lintas wilayah yang menggunakan peering VPC Amazon dan peering Transit Gateway secara otomatis dienkripsi massal saat keluar dari Wilayah. Lapisan enkripsi tambahan 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, M7i-flex
 - Komputasi dioptimalkan: C5a, C5ad, C5n, C6a, C6gn, C6i, C6iD, C6in, C7a, C7g, C7gd, C7gn, C7i, C7i-flex
 - Memori yang dioptimalkan: R5dn, R5n, R6a, R6i, R6iDn, R6in, R6iD, R7a, R7g, R7gd, R7i, R7iZ, U-3tb1, U-6tb1, U-9tb1, U-12tb1, U-24tb1, U7i-12TB, U7in-16TB Tb, U7in-24TB, U7in-32tb, X2idn, X2iEDN, X2IEZN
 - Penyimpanan dioptimalkan: D3, D3en, I3en, I4G, i4i, iM4GN, Is4gen
 - Komputasi yang dipercepat: DL1, DL2q, G4ad, G4dn, G5, G6, Gr6, Inf1, Inf2, P3dn, P4d, P4de, P5, Trn1, Trn1n, VT1

- Komputasi performa tinggi: Hpc6a, Hpc6id, Hpc7a, Hpc7g
- 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 enkripsi tambahan 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 tipe instans yang mengenkripsi lalu lintas dalam transit antar instans menggunakan AWS CLI

Gunakan perintah 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 AWS asalnya 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 Session Manager atau Run Command dienkripsi menggunakan TLS 1.2, dan permintaan untuk membuat koneksi ditandatangani menggunakan [SiGv4](#), dan diautentikasi serta diotorisasi 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 [Keamanan AWS Cloud](#). Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan](#).

Anda menggunakan panggilan API yang AWS dipublikasikan 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.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan prinsipal IAM. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

Untuk informasi selengkapnya, lihat [Perlindungan Infrastruktur](#) di Pilar Keamanan — AWS Well-Architected Framework.

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 instance 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 MAC dan alamat IP yang diberikan kepada mereka. Jika tidak, lalu lintas akan menurun.

Secara default, instans tidak dapat menerima lalu lintas yang tidak secara khusus ditujukan padanya. Jika Anda harus menjalankan layanan Network Address Translation (NAT), perutean, atau firewall pada instans Anda, maka Anda dapat menonaktifkan pemeriksaan sumber/tujuan untuk 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](#). Konfigurasi 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 instance 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. Juga, karena ACL jaringan berlaku untuk seluruh subnet, mereka dapat digunakan seolah-olah sebuah instance pernah diluncurkan defense-in-depth secara tidak sengaja tanpa grup keamanan yang benar.

- [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 yang aman seperti enkapsulasi RDP melalui SSL/TLS. Remote Desktop Gateway Quick Start menyediakan praktik terbaik untuk menerapkan gateway desktop jarak jauh, termasuk untuk mengonfigurasi RDP untuk menggunakan SSL/TLS.
- [Instans Windows] Gunakan Active Directory atau AWS Directory Service untuk mengontrol secara ketat dan terpusat dan memantau akses pengguna dan grup interaktif ke instance Windows, 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 ke subset PowerShell perintah Windows yang diperlukan untuk administrasi instance. Untuk informasi tambahan, lihat bagian "Mengelola Akses Tingkat OS pada Amazon EC2" dalam laporan resmi [Praktik Terbaik Keamanan AWS](#).
- [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. Sebagai gantinya, 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 ke server Windows.

- Lakukan konfigurasi pada tabel rute subnet Amazon VPC dengan rute jaringan minimal yang diperlukan. Misalnya, tempatkan hanya instans Amazon EC2 yang membalas 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 perubahan, mempermudah untuk meng-audit perubahan pada aturan grup keamanan atau skrip verifikasi aturan otomatis. Menggunakan beberapa antarmuka jaringan juga menyediakan opsi tambahan untuk mengontrol lalu lintas jaringan, termasuk kemampuan untuk membuat kebijakan perutean berbasis host atau memanfaatkan aturan perutean subnet VPC yang berbeda berdasarkan subnet yang ditetapkan dari antarmuka jaringan.
- Gunakan AWS Virtual Private Network atau AWS Direct Connect untuk membuat koneksi pribadi dari jaringan jarak jauh Anda ke VPC Anda. Untuk informasi selengkapnya, lihat [Opsi Konektivitas Network-to-Amazon VPC](#).
- Gunakan [Log Aliran VPC](#) untuk memantau lalu lintas yang menjangkau instans Anda.
- Gunakan [Perlindungan GuardDuty Malware](#) untuk mengidentifikasi perilaku mencurigakan yang menunjukkan 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 Runtime Monitoring](#) untuk mengidentifikasi dan merespons potensi ancaman terhadap instans Anda. Untuk informasi selengkapnya, lihat [Cara kerja Runtime Monitoring dengan instans Amazon EC2](#).
- Gunakan [AWS Security Hub](#), [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 [AWS Systems Manager Session Manager](#) 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 instance Anda.
- [Instans Windows] Banyak peran OS Windows dan aplikasi bisnis Microsoft juga menyediakan fungsionalitas yang ditingkatkan seperti pembatasan Rentang Alamat IP dalam IIS, kebijakan penyaringan TCP/IP 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 AWS 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 perintah [describe-availability-zones](#) untuk menampilkan 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 AMI di seluruh Wilayah
- Menyalin snapshot EBS di seluruh Wilayah
- Menerapkan otomatisasi AMI yang didukung EBS 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

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat [Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS](#) dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat [Program AWS Kepatuhan Program AWS](#).

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. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Memulai Cepat Keamanan dan Kepatuhan — Panduan](#) penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar AWS yang berfokus pada keamanan dan kepatuhan.
- [Arsitektur untuk Keamanan dan Kepatuhan HIPAA di Amazon Web Services](#) — Whitepaper ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat aplikasi yang memenuhi syarat HIPAA.

Note

Tidak semua memenuhi Layanan AWS syarat HIPAA. Untuk informasi selengkapnya, lihat [Referensi Layanan yang Memenuhi Syarat HIPAA](#).

- [AWS Sumber Daya AWS](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [AWS Panduan Kepatuhan Pelanggan](#) - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO)).
- [Mengevaluasi Sumber Daya dengan Aturan](#) dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— Ini Layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS. Security Hub menggunakan kontrol keamanan untuk sumber daya AWS Anda serta untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat [Referensi kontrol Security Hub](#).
- [Amazon GuardDuty](#) — Ini Layanan AWS mendeteksi potensi ancaman terhadap beban kerja Akun AWS, kontainer, dan data Anda dengan memantau lingkungan Anda untuk aktivitas yang mencurigakan dan berbahaya. GuardDuty dapat membantu Anda mengatasi berbagai persyaratan kepatuhan, seperti PCI DSS, dengan memenuhi persyaratan deteksi intrusi yang diamanatkan oleh kerangka kerja kepatuhan tertentu.
- [AWS Audit Manager](#)Ini Layanan AWS membantu Anda terus mengaudit AWS penggunaan Anda untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

Manajemen identitas dan akses untuk Amazon EC2

Kredensial keamanan Anda mengidentifikasi Anda ke layanan AWS dan memberi Anda penggunaan sumber daya tanpa batas, seperti AWS sumber daya Amazon EC2 Anda. Anda dapat menggunakan fitur Amazon EC2 dan AWS Identity and Access Management (IAM) untuk memungkinkan pengguna, layanan, dan aplikasi lain untuk menggunakan sumber daya Amazon EC2 Anda tanpa perlu membagikan kredensial keamanan Anda. Anda dapat menggunakan IAM untuk mengontrol cara pengguna lain menggunakan sumber daya di AWS akun Anda, dan Anda dapat menggunakan grup keamanan untuk mengontrol akses ke instans Amazon EC2 Anda. Anda dapat memilih untuk mengizinkan penggunaan sumber daya Amazon EC2 Anda secara penuh atau penggunaan secara terbatas.

Untuk praktik terbaik untuk mengamankan AWS sumber daya Anda menggunakan IAM, lihat [Praktik terbaik keamanan di IAM](#).

Daftar Isi

- [Akses jaringan ke instans Anda](#)
- [Atribut-atribut izin Amazon EC2](#)
- [IAM dan Amazon EC2](#)
- [Kebijakan IAM untuk Amazon EC2](#)
- [AWS kebijakan terkelola untuk Amazon EC2](#)
- [IAM role untuk Amazon EC2](#)

Akses jaringan ke instans Anda

Grup keamanan bertindak sebagai firewall yang mengendalikan lalu lintas yang diperbolehkan untuk mencapai satu atau beberapa instans. Saat Anda meluncurkan instans, artinya Anda menempatkan satu atau beberapa grup keamanan pada instans tersebut. Anda menambahkan aturan ke setiap grup keamanan yang mengendalikan lalu lintas untuk instans tersebut. Anda dapat melakukan modifikasi terhadap aturan untuk grup keamanan kapan saja; aturan baru tersebut secara otomatis akan diterapkan pada semua instans tempat grup keamanan ditetapkan.

Untuk informasi selengkapnya, lihat [Aturan-aturan grup keamanan](#).

Atribut-atribut izin Amazon EC2

Organisasi Anda mungkin memiliki beberapa AWS akun. Amazon EC2 memungkinkan Anda menentukan AWS akun tambahan yang dapat menggunakan Amazon Machine Images (AMI) dan snapshot Amazon EBS Anda. Izin ini hanya berfungsi pada tingkat AWS akun; Anda tidak dapat membatasi izin untuk pengguna tertentu dalam akun yang ditentukan. AWS Semua pengguna dalam akun AWS yang telah Anda tentukan tersebut dapat menggunakan AMI atau snapshot.

Setiap AMI memiliki atribut `LaunchPermission` yang mengendalikan akun AWS mana yang dapat mengakses AMI. Untuk informasi selengkapnya, lihat [Menjadikan AMI publik](#).

Setiap snapshot Amazon EBS memiliki `createVolumePermission` atribut yang mengontrol AWS akun mana yang dapat menggunakan snapshot. Untuk informasi selengkapnya, lihat [Membagikan snapshot Amazon EBS](#) di Panduan Pengguna Amazon EBS.

IAM dan Amazon EC2

IAM memungkinkan Anda untuk melakukan hal berikut:

- Buat pengguna dan grup di bawah Akun AWS
- Tetapkan kredensial keamanan unik untuk setiap pengguna di bawah Akun AWS
- Kontrol izin setiap pengguna untuk melakukan tugas menggunakan sumber daya AWS
- Izinkan pengguna di tempat lain Akun AWS untuk berbagi AWS sumber daya Anda
- Buat peran untuk Anda Akun AWS dan tentukan pengguna atau layanan yang dapat mengasumsikan mereka
- Gunakan identitas yang ada untuk perusahaan Anda untuk memberikan izin untuk melakukan tugas menggunakan sumber daya AWS

Dengan menggunakan IAM dengan Amazon EC2, Anda dapat mengendalikan apakah para pengguna dalam organisasi Anda dapat melakukan tugas menggunakan tindakan API Amazon EC2 tertentu dan apakah mereka dapat menggunakan sumber daya AWS tertentu.

Topik ini akan membantu Anda menjawab pertanyaan-pertanyaan berikut:

- Bagaimana cara saya membuat grup dan pengguna dalam IAM?
- Bagaimana cara saya membuat kebijakan?
- Kebijakan IAM apa yang saya perlukan untuk menjalankan tugas-tugas dalam Amazon EC2?
- Bagaimana cara saya memberikan izin untuk melakukan tindakan-tindakan dalam Amazon EC2?
- Bagaimana cara saya memberikan izin untuk melakukan tindakan-tindakan pada sumber daya tertentu dalam Amazon EC2?

Membuat pengguna, grup, dan peran

Anda dapat membuat pengguna dan grup untuk Anda Akun AWS dan kemudian menetapkan mereka izin yang mereka butuhkan. Sebagai praktik terbaik, pengguna harus memperoleh izin dengan mengambil peran IAM.

Sebuah [peran IAM](#) adalah identitas IAM yang dapat Anda buat di akun yang memiliki izin tertentu. Peran IAM mirip dengan pengguna IAM karena merupakan AWS identitas dengan kebijakan izin yang menentukan apa yang dapat dan tidak dapat dilakukan identitas. AWS Namun, alih-alih secara

unik terkait dengan satu orang, peran dimaksudkan untuk menjadi dapat diambil oleh siapa pun yang membutuhkannya. Selain itu, peran tidak memiliki kredensial jangka panjang standar seperti kata sandi atau kunci akses yang terkait dengannya. Sebagai gantinya, saat Anda mengambil peran, peran tersebut akan memberikan kredensial keamanan sementara untuk sesi peran. Untuk informasi selengkapnya tentang cara membuat peran IAM dan memberikan izin dengannya, lihat [the section called “IAM role”](#)

Topik terkait

Untuk informasi selengkapnya tentang IAM, lihat berikut ini:

- [Kebijakan IAM untuk Amazon EC2](#)
- [IAM role untuk Amazon EC2](#)
- [AWS Identity and Access Management \(IAM\)](#)
- [Panduan Pengguna IAM](#)

Kebijakan IAM 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 kustom, lihat [Mengelola Kebijakan IAM](#).

Memulai

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.

Amazon EC2 mendukung izin tingkat sumber daya secara parsial. Artinya untuk beberapa tindakan EC2 API, Anda tidak dapat menentukan sumber daya mana yang diizinkan untuk digunakan oleh seorang pengguna untuk tindakan tersebut. Sebaliknya, Anda harus mengizinkan para pengguna untuk menggunakan semua sumber daya untuk tindakan tersebut.

Tugas	Topik
Memahami struktur dasar kebijakan	Sintaksis kebijakan
Menentukan tindakan dalam kebijakan Anda	Tindakan-tindakan untuk Amazon EC2
Menentukan sumber daya tertentu dalam kebijakan Anda	Amazon Resource Name (ARN) untuk Amazon EC2
Menerapkan syarat terhadap penggunaan sumber daya	Kunci syarat untuk Amazon EC2
Cara menggunakan izin tingkat sumber daya yang tersedia untuk Amazon EC2	Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2
Uji kebijakan Anda	Memeriksa apakah pengguna memiliki izin yang diperlukan
Membuat kebijakan IAM	Membuat kebijakan berdasarkan aktivitas akses
Contoh kebijakan untuk CLI atau SDK	Contoh kebijakan untuk bekerja dengan AWS CLI atau AWS SDK
Contoh kebijakan untuk konsol Amazon EC2	Kebijakan contoh yang digunakan dalam konsol Amazon EC2

Berikan izin kepada pengguna, grup, dan peran

Berikut ini adalah contoh dari beberapa kebijakan AWS terkelola yang tersedia untuk digunakan jika memenuhi kebutuhan Anda:

- `PowerUserAccess`
- `ReadOnlyAccess`

- `AmazonEC2FullAccess`
- `AmazonEC2ReadOnlyAccess`

Untuk informasi selengkapnya, lihat [the section called “AWS kebijakan terkelola”](#).

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.

- (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Struktur kebijakan

Topik-topik berikut ini menjelaskan struktur dari kebijakan IAM.

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](#)
- [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 bekerja dengan AWS CLI atau AWS SDK](#).

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 [Kebijakan IAM 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

Izin tingkat sumber daya mengacu pada kemampuan untuk menentukan sumber daya mana yang boleh digunakan oleh para pengguna 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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#).

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 menetapkan 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.

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.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ec2:AuthorizeSecurityGroupIngress",
      "ec2:AuthorizeSecurityGroupEgress",
      "ec2:RevokeSecurityGroupIngress",
      "ec2:RevokeSecurityGroupEgress"],
    "Resource": "arn:aws:ec2:region:account:security-group/*",
    "Condition": {
      "StringEquals": {
        "ec2:Vpc": "arn:aws:ec2:region:account: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.

Semua tindakan Amazon EC2 mendukung kunci syarat `aws:RequestedRegion` dan `ec2:Region`. Untuk informasi selengkapnya, lihat [Contoh: Membatasi akses ke suatu Wilayah tertentu](#).

Kunci syarat `ec2:SourceInstanceARN`

Kunci syarat `ec2:SourceInstanceARN` dapat digunakan untuk syarat-syarat yang menentukan ARN dari instans tempat permintaan dibuat. Ini adalah kunci kondisi AWS global dan tidak spesifik layanan. Untuk contoh kebijakan, lihat [Amazon EC2: Melampirkan atau melepaskan lampiran volume ke instans EC2](#) dan [Contoh: Izinkan instance tertentu untuk melihat sumber daya di AWS layanan lain](#). Kunci `ec2:SourceInstanceARN` ini tidak dapat digunakan sebagai variabel untuk mengisi ARN untuk elemen Resource di pernyataan.

Untuk contoh pernyataan kebijakan untuk Amazon EC2, lihat [Contoh kebijakan untuk bekerja dengan AWS CLI atau AWS SDK](#).

Kunci syarat `ec2:Attribute`

Kunci syarat `ec2:Attribute` dapat digunakan untuk syarat-syarat yang memfilter akses berdasarkan atribut sumber daya. Kunci kondisi hanya mendukung properti yang bertipe data primitif (seperti string atau integer), atau [AttributeValue](#) objek kompleks yang hanya memiliki properti Value (seperti Deskripsi atau `ImdsSupport` objek aksi API [ModifyImageAtribut](#)).

Important

Kunci kondisi tidak dapat digunakan dengan objek kompleks yang memiliki beberapa properti, seperti `LaunchPermission` objek aksi API [ModifyImageAtribut](#).

Misalnya, kebijakan berikut menggunakan kunci `ec2:Attribute/Description` kondisi untuk memfilter akses berdasarkan objek `Description` kompleks dari tindakan API `ModifyImageAttribute`. Kunci syarat hanya mengizinkan permintaan yang memodifikasi deskripsi citra ke `Production` atau `Development`.

```
{
  "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 `Attribute` primitif dari tindakan API `ModifyImageAttribute`. Kunci syarat menolak semua permintaan yang berusaha memodifikasi deskripsi citra.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:ModifyImageAttribute",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:Attribute": "Description"
        }
      }
    }
  ]
}
```

}

Kunci syarat **ec2:ResourceID**

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 `CopyImageAPI`, maka Anda harus menentukan kunci kondisi sebagai berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:ImageID": "*"
        }
      }
    }
  ]
}
```

Kunci syarat	Tindakan API			
<code>ec2:DhcpOptionsID</code>	<ul style="list-style-type: none"> CreateDhcpPilihan 			
<code>ec2:ImageID</code>	<ul style="list-style-type: none"> CopyImage CreateImage 			

Kunci syarat	Tindakan API			
	<ul style="list-style-type: none"> ImportImage RegisterImage 			
ec2:InstanceID	<ul style="list-style-type: none"> RunInstances ImportInstance 			
ec2:InternetGatewayID	<ul style="list-style-type: none"> CreateInternetGerbang 			
ec2:NetworkACLID	<ul style="list-style-type: none"> CreateNetworkAcl 			
ec2:NetworkInterfaceID	<ul style="list-style-type: none"> CreateNetworkAntarmuka 			
ec2:PlacementGroupName	<ul style="list-style-type: none"> CreatePlacementKelompok 			
ec2:RouteTableID	<ul style="list-style-type: none"> CreateRouteTabel 			

Kunci syarat	Tindakan API			
ec2:SecurityGroupID	<ul style="list-style-type: none">CreateSecurityKelompok			
ec2:SnapshotID	<ul style="list-style-type: none">CopySnapshotCreateSnapshotCreateSnapshotsImportSnapshots			
ec2:SubnetID	<ul style="list-style-type: none">CreateSubnet			
ec2:VolumeID	<ul style="list-style-type: none">CreateVolumeImportVolume			
ec2:VpcID	<ul style="list-style-type: none">CreateVpc			
ec2:VpcPeeringConnectionID	<ul style="list-style-type: none">CreateVpcPeeringConnection			

Kami menyarankan agar Anda menghindari penggunaan kunci syarat `ec2:ResourceID` dengan tindakan API ini. Sebagai gantinya, jika Anda perlu memfilter akses berdasarkan ID sumber daya tertentu, sebaiknya Anda melakukannya menggunakan elemen kebijakan `Resource`, seperti berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-01234567890abcdef"
    }
  ]
}
```

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 decode pada pesan tersebut menggunakan tindakan `DecodeAuthorizationMessage`. Untuk informasi selengkapnya, lihat [DecodeAuthorizationPesan](#) di Referensi AWS Security Token Service API, dan pesan [decode-authorization-di](#) Referensi Perintah.AWS CLI

Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat

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 [Mengendalikan akses ke sumber daya EC2 menggunakan tanda sumber daya](#).

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:region:account:*/*",
    "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:region:account:*/*",
      "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 bekerja dengan AWS CLI atau AWS SDK](#). Untuk informasi selengkapnya, lihat [Membuat Syarat yang Menguji Beberapa Nilai Kunci](#) dalam Panduan Pengguna IAM.

Mengendalikan akses ke sumber daya EC2 menggunakan tanda sumber daya

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 bekerja dengan AWS CLI atau AWS SDK](#).

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.

Contoh kebijakan untuk bekerja dengan AWS CLI atau AWS SDK

Anda harus memberi pengguna izin yang mereka perlukan untuk Amazon EC2 menggunakan kebijakan IAM. Contoh berikut menunjukkan pernyataan kebijakan yang dapat Anda gunakan untuk mengontrol izin yang dimiliki oleh pengguna pada Amazon EC2. Kebijakan ini dirancang untuk permintaan yang dibuat dengan AWS CLI atau AWS SDK. Untuk informasi selengkapnya, lihat [Membuat kebijakan IAM](#) dalam Panduan Pengguna IAM. Untuk contoh kebijakan yang digunakan dalam konsol Amazon EC2, lihat [Kebijakan contoh yang digunakan dalam konsol Amazon EC2](#). Untuk contoh kebijakan IAM tertentu untuk Amazon VPC, lihat [Manajemen Identitas dan Akses untuk Amazon VPC](#).

Dalam contoh-contoh berikut, ganti setiap *placeholder input pengguna* dengan informasi Anda sendiri.

Contoh

- [Contoh: Akses hanya-baca](#)
- [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: Izinkan instance tertentu untuk melihat sumber daya di AWS layanan lain](#)
- [Contoh: Cara menggunakan templat peluncuran](#)
- [Cara menggunakan metadata instans](#)
- [Bekerja dengan volume dan snapshot Amazon EBS](#)

Contoh: Akses hanya-baca

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.

```
{
  "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.

```
{
  "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.

```
{
  "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.

```
{
  "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 para pengguna untuk mendeskripsikan semua instans, memulai dan menghentikan instans i-1234567890abcdef0 dan i-0598c7d356eba48d7 saja, dan untuk mengakhiri instans di Wilayah AS Timur (Virginia Utara) (us-east-1) yang memiliki tanda sumber daya "purpose=test" saja.

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 Wilayah AS Timur (Virginia N.us-east-1) yang termasuk dalam AWS akun yang ditentukan, tetapi hanya jika instance memiliki tag. "purpose=test" Elemen Condition memenuhi syarat ketika pernyataan kebijakan berlaku.

```
{
  "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:account-id:instance/i-1234567890abcdef0",
    "arn:aws:ec2:us-east-1:account-id:instance/i-0598c7d356eba48d7"
  ]
},
{
  "Effect": "Allow",
  "Action": "ec2:TerminateInstances",
  "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceTag/purpose": "test"
    }
  }
}
]
}

```

Luncurkan instance () RunInstances

Tindakan [RunInstances](#) API meluncurkan satu atau beberapa Instans Sesuai Permintaan 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. Peluncuran dari AMI yang didukung Amazon EBS akan 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).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region::image/ami-9e1670f7",
        "arn:aws:ec2:region::image/ami-45cf5c3c",
        "arn:aws:ec2:region:account-id:instance/*",
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:key-pair/*",
        "arn:aws:ec2:region:account-id:security-group/*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id: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).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region::image/ami-*"
      ],
      "Condition": {
        "StringEquals": {
          "ec2:Owner": "amazon"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:account-id:instance/*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:key-pair/*",
        "arn:aws:ec2:region:account-id: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`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
        "StringEquals": {
          "ec2:InstanceType": ["t2.micro", "t2.small"]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region::image/ami-*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:key-pair/*",
        "arn:aws:ec2:region:account-id: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`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
```

```

    "StringNotEquals": {
      "ec2:InstanceType": ["t2.micro", "t2.small"]
    }
  },
  {
    "Effect": "Allow",
    "Action": "ec2:RunInstances",
    "Resource": [
      "arn:aws:ec2:region::image/ami-*",
      "arn:aws:ec2:region:account-id:network-interface/*",
      "arn:aws:ec2:region:account-id:instance/*",
      "arn:aws:ec2:region:account-id:subnet/*",
      "arn:aws:ec2:region:account-id:volume/*",
      "arn:aws:ec2:region:account-id:key-pair/*",
      "arn:aws:ec2:region:account-id: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).

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:account-id:subnet/subnet-12345678",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:instance/*",
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region::image/ami-*",
        "arn:aws:ec2:region:account-id:key-pair/*",
        "arn:aws:ec2:region:account-id: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.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:account-id:network-interface/*"
      ],
      "Condition": {
        "ArnNotEquals": {
          "ec2:Subnet": "arn:aws:ec2:region:account-id:subnet/subnet-12345678"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region::image/ami-*",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:instance/*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:key-pair/*",
        "arn:aws:ec2:region:account-id: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.

```
{
  "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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:region::image/*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:security-group/*",
        "arn:aws:ec2:region:account-id:key-pair/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/environment": "production" ,
          "aws:RequestTag/purpose": "webserver"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "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.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:region:image/*",
        "arn:aws:ec2:region:account-id:subnet/*",
        "arn:aws:ec2:region:account-id:network-interface/*",
        "arn:aws:ec2:region:account-id:security-group/*",
        "arn:aws:ec2:region:account-id:key-pair/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:region:account-id:volume/*",
        "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:TagKeys": ["environment", "webserver"]
        }
      }
    }
  ]
}

```

```

    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "ec2:CreateTags"
  ],
  "Resource": "arn:aws:ec2:region:account-id:*/*",
  "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`.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "Condition": {

```

```

    "StringEquals": {
      "aws:RequestTag/purpose": "test",
      "ec2:CreateAction" : "RunInstances"
    },
    "ForAllValues:StringEquals": {
      "aws:TagKeys": "purpose"
    }
  }
}
]
}

```

Untuk melarang siapa pun yang dipanggil tag di create for RunInstances

```

{
  "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": "VisualEditor0",
      "Effect": "Deny",
      "Action": "ec2:CreateTags",
      "Resource": "*"
    }
  ]
}

```

Hanya izinkan tag tertentu untuk spot-instances-request. Inkonsistensi kejutan nomor 2 akan mempengaruhi hasilnya. Dalam keadaan normal, tidak menentukan tanda akan menghasilkan Tidak terautentikasi. Dalam hal ini spot-instances-request, kebijakan ini tidak akan dievaluasi jika tidak ada spot-instances-request tag, sehingga permintaan Spot on Run non-tag akan berhasil.

```
{
  "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": "VisualEditor0",
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/environment": "production"
        }
      }
    }
  ]
}
```

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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d310e"
        },
        "Bool": {
          "ec2:IsLaunchTemplateResource": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id: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`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances"
      ],
      "Resource": [
        "arn:aws:ec2:*:account-id: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:*:account-id:network-interface/*",
        "arn:aws:ec2:*:account-id:instance/*",
        "arn:aws:ec2:*:account-id:subnet/*",
        "arn:aws:ec2:*:account-id:volume/*",
        "arn:aws:ec2:*:account-id:key-pair/*",
        "arn:aws:ec2:*:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id: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 dari pernyataan menggunakan [NotResource](#) elemen untuk memungkinkan semua sumber daya lain 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.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "NotResource": ["arn:aws:ec2:region:account-id:subnet/*",
                     "arn:aws:ec2:region:account-id:network-interface/*" ],
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": ["arn:aws:ec2:region:account-id:subnet/*",
                   "arn:aws:ec2:region:account-id:network-interface/*" ],
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id: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`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "NotResource": "arn:aws:ec2:region:account-id:launch-template/*",
      "Condition": {
        "ArnLike": {
          "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
        },
        "Bool": {
          "ec2:IsLaunchTemplateResource": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:region:account-id: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 Instans Spot saat membuat. Sumber daya yang akan ditentukan `RunInstances` adalah `spot-instances-request`.

Sumber daya `spot-instances-request` dievaluasi dalam kebijakan IAM sebagaimana berikut ini:

- Jika Anda tidak menandai permintaan Instans Spot saat membuat, Amazon EC2 tidak mengevaluasi `spot-instances-request` sumber daya dalam pernyataan. `RunInstances`
- Jika Anda menandai permintaan Instans Spot saat membuat, Amazon EC2 akan mengevaluasi `spot-instances-request` sumber daya dalam pernyataan. `RunInstances`

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 membuat, Anda tidak perlu secara eksplisit mengizinkan `spot-instances-request` sumber daya; panggilan akan berhasil.
- Jika Anda menggunakan `RunInstances` untuk membuat permintaan Instans Spot dan bermaksud untuk menandai permintaan Instans Spot saat membuat, 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 bermaksud untuk menandai permintaan Instans Spot saat membuat, Anda harus menentukan `spot-instances-request` sumber daya atau `*` wildcard dalam pernyataan `CreateTags allow`, jika tidak panggilan akan gagal.

Anda dapat meminta Instans Spot menggunakan `RunInstances` atau `RequestSpotInstances`. Contoh berikut kebijakan IAM hanya berlaku saat meminta Instans Spot menggunakan `RunInstances`

Contoh: Minta Instans Spot menggunakan `RunInstances`

Kebijakan berikut memungkinkan pengguna untuk meminta Instans Spot dengan menggunakan `RunInstances` tindakan. 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 membuat. Ini karena Amazon

EC2 tidak mengevaluasi `spot-instances-request` sumber daya dalam `RunInstances` pernyataan jika permintaan Instans Spot tidak ditandai pada `create`.

```
{
  "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 ini dimaksudkan untuk memberikan izin kepada para pengguna untuk meluncurkan Instans Sesuai Permintaan, tetapi menolak memberikan izin untuk permintaan Instans Spot. `spot-instances-request` Sumber 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 ditandai pada `create`.

```
{
  "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": "DenySpotInstancesRequests - NOT SUPPORTED - DO NOT USE!",
      "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. `spot-instances-request` Sumber 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 membuat, Amazon EC2 akan mengevaluasi `spot-instances-request` sumber daya dalam pernyataan. `RunInstances` Oleh karena itu, Anda harus secara eksplisit mengizinkan `spot-instances-request` sumber daya untuk `RunInstances` tindakan tersebut, jika tidak panggilan akan gagal.

```
{
  "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. `spot-instances-request` Sumber 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 ditandai pada `create`, RunInstances panggilan akan gagal.

```
{
  "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 RunInstances izin untuk membuat permintaan Instans Spot hanya jika permintaan ditandai 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 menghasilkan permintaan Instans Spot yang tidak diberi tag dibuat oleh RunInstances. Perhatikan bahwa menentukan tag lain selain `environment=production` menghasilkan Unauthenticated kesalahan, karena jika pengguna menandai permintaan Instans Spot, Amazon EC2 mengevaluasi sumber daya `spot-instances-request` dalam pernyataan RunInstances.

```
{
  "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": "RequestSpotInstancesOnlyIfTagIs_environment=production - NOT SUPPORTED - DO NOT USE!",
      "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 RunInstances izin untuk membuat permintaan Instans Spot jika permintaan tersebut ditandai 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.

```

{
  "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.

```
{
  "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.

```
{
  "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 tanda lain yang diizinkan dan pengguna tidak dapat memberi tanda pada tipe sumber daya lainnya.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:DeleteTags",
      "Resource": "arn:aws:ec2:us-east-1:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {

```

```

    "Effect": "Allow",
    "Action": [
      "ec2:DeleteTags"
    ],
    "Resource": "arn:aws:ec2:region:account-id:*/*",
    "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 tindakan `iam:PassRole` guna meneruskan peran ke instans.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:AssociateIamInstanceProfile",
        "ec2:ReplaceIamInstanceProfileAssociation",
        "ec2:DisassociateIamInstanceProfile"
      ],
      "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/department": "test"
        }
      }
    }
  ]
}

```

```

    }
  },
  {
    "Effect": "Allow",
    "Action": "ec2:DescribeIamInstanceProfileAssociations",
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::account-id: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](#).

```

{
  "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::account-id: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.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteRoute",
        "ec2:CreateRoute",
        "ec2:ReplaceRoute"
      ],
      "Resource": [
        "arn:aws:ec2:region:account-id:route-table/*"
      ],
      "Condition": {
        "StringEquals": {
          "ec2:Vpc": "arn:aws:ec2:region:account-id:vpc/vpc-ec43eb89"
        }
      }
    }
  ]
}

```

Contoh: Izinkan instance tertentu untuk melihat sumber daya di AWS layanan lain

Berikut ini adalah contoh kebijakan yang dapat Anda lampirkan ke IAM role. Kebijakan ini memungkinkan instance untuk melihat sumber daya di berbagai AWS layanan. Kebijakan ini menggunakan kunci syarat `ec2:SourceInstanceARN` untuk menentukan bahwa instans tempat permintaan dibuat harus instans `i-093452212644b0dd6`. Jika IAM role yang sama dikaitkan dengan instans yang lain, maka instans lain tersebut tidak akan dapat melakukan tindakan apa pun.

`ec2:SourceInstanceARN` kuncinya adalah kunci kondisi AWS global, oleh karena itu dapat digunakan untuk tindakan layanan lainnya, bukan hanya Amazon EC2.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVolumes",
        "s3:ListAllMyBuckets",
        "dynamodb:ListTables",
        "rds:DescribeDBInstances"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "ArnEquals": {
          "ec2:SourceInstanceARN": "arn:aws:ec2:region:account-id: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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:CreateLaunchTemplateVersion",
        "ec2:ModifyLaunchTemplate"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:region:account-id: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`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:DeleteLaunchTemplate",
        "ec2:DeleteLaunchTemplateVersions"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:region:account-id: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. `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 [LaunchTemplateSpesifikasi](#) di Referensi API Auto Scaling Amazon EC2, khususnya parameter `Version`.
4. Untuk grup Auto Scaling yang menggunakan konfigurasi peluncuran, ganti konfigurasi peluncuran itu dengan templat peluncuran. Untuk informasi selengkapnya, lihat [Mengganti Konfigurasi Peluncuran dengan Templat Peluncuran](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.
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](#) di Referensi AWS CLI Perintah.

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 memilih untuk meminta penggunaan IMDSv2 (ditunjukkan oleh).

`"ec2:MetadataHttpTokens": "required"` Jika Anda tidak menentukan bahwa instance memerlukan `ImDSv2`, Anda mendapatkan `UnauthorizedOperation` kesalahan saat memanggil API. `RunInstances`

```
{  
  "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`.

```

{
  "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.

```
{
  "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-iamds-admins` untuk melakukan perubahan pada opsi metadata instans. Jika ada prinsipal selain `ec2-iamds-admins` peran yang mencoba memanggil `ModifyInstanceMetadataOptions` API, itu akan mendapatkan `UnauthorizedOperation` kesalahan. Pernyataan ini dapat digunakan untuk mengontrol penggunaan `ModifyInstanceMetadataOptions` API; saat ini tidak ada kontrol akses (kondisi) berbutir halus untuk API. `ModifyInstanceMetadataOptions`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowOnlyIamdsAdminsToModifySettings",
      "Effect": "Deny",
      "Action": "ec2:ModifyInstanceMetadataOptions",
      "Resource": "*",
      "Condition": {
        "StringNotLike": {
```

```

    "aws:PrincipalARN": "arn:aws:iam::*:role/ec2-imsd-admins"
  }
}
]
}

```

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 `UnauthorizedOperation`. Pernyataan/kebijakan ini dapat diterapkan secara umum karena, jika permintaan tidak ditandatangani oleh kredensial peran EC2, maka tidak ada dampak yang terjadi.

```

{
  "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 kebijakan berbasis identitas](#) untuk Amazon EBS.

Kebijakan contoh yang digunakan dalam konsol Amazon EC2

Anda harus memberi pengguna izin yang mereka perlukan untuk Amazon EC2 menggunakan kebijakan IAM. Anda dapat menggunakan kebijakan IAM untuk memberikan izin kepada para

pengguna untuk menampilkan dan menggunakan sumber daya tertentu dalam konsol Amazon EC2. Anda dapat menggunakan contoh kebijakan di bagian sebelumnya; namun, kebijakan tersebut dirancang untuk permintaan yang dibuat dengan AWS CLI atau AWS SDK. Untuk informasi selengkapnya, lihat [Contoh kebijakan untuk bekerja dengan AWS CLI atau AWS SDK](#) dan [Membuat kebijakan IAM](#) dalam 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 postingan Blog AWS Keamanan berikut: [Memberikan Izin kepada Pengguna untuk Bekerja di Konsol Amazon EC2](#).

Tip

Untuk membantu Anda mengetahui tindakan API mana yang dibutuhkan untuk melakukan tugas-tugas dalam konsol, Anda dapat menggunakan layanan seperti AWS CloudTrail. Untuk informasi selengkapnya, lihat [Panduan Pengguna 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-authorization-message](#) di AWS CLI

Contoh

- [Contoh: Akses hanya-baca](#)
- [Contoh: Menggunakan wizard peluncuran instans EC2](#)
- [Contoh: Cara menggunakan grup keamanan](#)
- [Contoh: Cara menggunakan alamat IP Elastis](#)
- [Contoh: Cara Menggunakan Instans Cadangan](#)

Contoh: Akses hanya-baca

Untuk memungkinkan para pengguna menampilkan semua sumber daya dalam konsol Amazon EC2, Anda dapat menggunakan kebijakan yang sama seperti contoh berikut ini: [Contoh: Akses hanya-](#)

[baca](#). 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.

```
{
  "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](#).

Lihat contoh dan metrik CloudWatch

Kebijakan berikut memungkinkan pengguna untuk melihat instans di konsol Amazon EC2, CloudWatch serta alarm dan metrik di tab Pemantauan halaman Instans. Konsol Amazon EC2 menggunakan CloudWatch API untuk menampilkan alarm dan metrik, jadi Anda harus memberi pengguna izin untuk menggunakan `cloudwatch:DescribeAlarms`, `cloudwatch:DescribeAlarmsForMetric`, `cloudwatch:ListMetrics`, `cloudwatch:GetMetricStatistics` dan tindakan `cloudwatch:GetMetricData`

```
{
  "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.

```
{
  "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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#). 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 memberikan izin kepada para pengguna untuk menggunakan tindakan `ec2:CreateTags`, mereka dapat mengubah tanda dari sumber daya untuk menembus pembatasan-pembatasan tersebut. Untuk informasi selengkapnya, lihat [Mengendalikan akses ke sumber daya EC2 menggunakan tanda sumber daya](#).

- 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 sa-east-1. 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.

```
{
  "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:sa-east-1:111122223333:network-interface/*",
    "arn:aws:ec2:sa-east-1:111122223333:volume/*",
    "arn:aws:ec2:sa-east-1:111122223333:key-pair/*",
    "arn:aws:ec2:sa-east-1:111122223333:security-group/*",
    "arn:aws:ec2:sa-east-1:111122223333:subnet/subnet-1a2b3c4d"
  ]
},
{
  "Effect": "Allow",
  "Action": "ec2:RunInstances",
  "Resource": [
    "arn:aws:ec2:sa-east-1:111122223333:instance/*"
  ],
  "Condition": {
    "StringEquals": {
      "ec2:InstanceType": "t2.micro"
    }
  }
},
{
  "Effect": "Allow",
  "Action": "ec2:RunInstances",
  "Resource": [
    "arn:aws:ec2:sa-east-1::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-*`.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ssm:DescribeParameters"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "ssm:GetParameters"
    ],
    "Resource": "arn:aws:ssm:us-east-2:123456123: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.

```
{
  "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:region:111122223333:security-group/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/Department": "Test"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:ModifySecurityGroupRules"
    ],
    "Resource": [
        "arn:aws:ec2:region: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`.

```
{
  "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:region:111122223333:security-group/*",
    "Condition": {
      "ArnEquals": {
        "ec2:Vpc": "arn:aws:ec2:region: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.

```
{
  "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 AWS Management Console.

Kebijakan ini akan memungkinkan para pengguna untuk menampilkan semua Instans Cadangan, serta Instans Sesuai Permintaan, dalam akun tersebut. Pengaturan izin tingkat sumber daya untuk masing-masing Instans Cadangan tidak dapat dilakukan.

```
{
  "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 AWS terkelola kami. Kebijakan ini mencakup kasus penggunaan umum dan tersedia di AWS akun Anda. Untuk informasi selengkapnya tentang kebijakan AWS [AWS terkelola](#), lihat [kebijakan terkelola](#) di Panduan Pengguna IAM.

AWS layanan memelihara dan memperbarui kebijakan AWS terkelola. Anda tidak dapat mengubah izin dalam kebijakan AWS terkelola. 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 AWS terkelola, sehingga pembaruan kebijakan tidak akan merusak izin yang ada.

Selain itu, AWS mendukung kebijakan terkelola untuk fungsi pekerjaan yang mencakup beberapa layanan. Misalnya, kebijakan AWS terkelola `ReadOnlyAccess` menyediakan akses hanya-baca ke semua AWS layanan dan sumber daya. Saat layanan meluncurkan fitur baru, AWS tambahkan izin hanya-baca 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 terkelola: `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 terkelola: `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 terkelola: `AWSEC2CapacityReservationFleetRolePolicy`

Kebijakan ini dilampirkan pada peran tertaut layanan yang bernama `AWSServiceRoleForEC2CapacityReservationFleet` untuk memungkinkan Reservasi Kapasitas untuk membuat, memodifikasi, dan membatalkan Reservasi Kapasitas atas nama Anda. Untuk informasi selengkapnya, lihat [Peran terkait layanan untuk Armada Reservasi Kapasitas](#).

Untuk melihat izin kebijakan ini, lihat [AWSEC2CapacityReservationFleetRolePolicy](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: 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 [Peran tertaut layanan untuk Armada EC2](#).

Untuk melihat izin kebijakan ini, lihat [AWSEC2FleetServiceRolePolicy](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: 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 [Peran tertaut layanan untuk Armada Spot](#).

Untuk melihat izin kebijakan ini, lihat [AWSEC2SpotFleetServiceRolePolicy](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: 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 [Peran tertaut layanan untuk permintaan Instans Spot](#).

Untuk melihat izin kebijakan ini, lihat [AWSEC2SpotServiceRolePolicy](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSEC2VssSnapshotPolicy

Anda dapat melampirkan kebijakan terkelola ini ke peran profil instans IAM yang Anda gunakan untuk instans Windows Amazon EC2. 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 terkelola: EC2FastLaunchFullAccess

Anda dapat melampirkan EC2FastLaunchFullAccess kebijakan ke profil instans Anda 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 dapat mengaktifkan atau menonaktifkan Peluncuran Cepat EC2, dan menggambarkan gambar Peluncuran Cepat EC2.
- Amazon EC2 — Akses diberikan untuk Amazon RunInstances EC2CreateTags , dan Jelaskan tindakan yang diperlukan untuk memverifikasi izin sumber daya.
- IAM — Akses diberikan untuk mendapatkan dan menggunakan profil instance yang namanya berisi ec2fastlaunch untuk membuat peran EC2FastLaunchServiceRolePolicy terkait layanan.

Untuk melihat izin kebijakan ini, lihat [EC2FastLaunchFullAccess](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: EC2FastLaunchServiceRolePolicy

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForEC2FastLaunch untuk memungkinkan Amazon EC2 membuat dan mengelola sekumpulan snapshot yang telah disediakan sebelumnya yang mengurangi waktu yang diperlukan untuk meluncurkan instans dari AMI berkemampuan Peluncuran Cepat EC2 Anda. Untuk informasi selengkapnya, lihat [the section called “Peran terkait layanan”](#).

Untuk melihat izin kebijakan ini, lihat [EC2FastLaunchServiceRolePolicy](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: Ec2InstanceConnectEndpoint

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForEC2InstanceConnect untuk mengizinkan Titik Akhir Koneksi Instans EC2 melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat [Peran tertaut layanan untuk EC2 Instance Connect Endpoint](#).

Untuk melihat izin kebijakan ini, lihat [Ec2InstanceConnectEndpoint](#) di Referensi Kebijakan AWS Terkelola.

Pembaruan Amazon EC2 ke AWS kebijakan terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk Amazon EC2 sejak layanan ini mulai melacak perubahan ini.

Perubahan	Deskripsi	Tanggal
EC2FastLaunchFullAccess – Kebijakan baru	Amazon EC2 menambahkan kebijakan ini untuk melakukan tindakan API yang terkait dengan fitur Peluncuran Cepat EC2 dari instans. Kebijakan ini dapat dilampirkan ke profil instans untuk instance yang diluncurkan dari AMI yang diaktifkan Peluncuran Cepat EC2.	14 Mei 2024
AWSEC2VssSnapshotPolicy – Kebijakan baru	Amazon EC2 menambahkan <code>AWSEC2VssSnapshotPolicy</code> kebijakan yang berisi izin untuk membuat dan menambahkan tag ke Amazon Machine Images (AMI) dan EBS Snapshots.	Maret 28, 2024
EC2FastLaunchServiceRolePolicy – Kebijakan baru	Amazon EC2 menambahkan fitur Peluncuran Cepat EC2 untuk memungkinkan AMI Windows meluncurkan instans lebih cepat dengan membuat serangkaian snapshot yang telah disediakan sebelumnya.	26 November 2021
Amazon EC2 mulai melakukan pelacakan perubahan	Amazon EC2 mulai melacak perubahan pada kebijakan yang dikelola AWS	1 Maret 2021

IAM role untuk Amazon EC2

Aplikasi harus menandatangani permintaan API mereka dengan AWS kredensialnya. 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 kredensial secara aman ke setiap instans, terutama yang AWS dibuat atas nama Anda, seperti Instans Spot atau instans di grup Auto Scaling. Anda juga harus dapat memperbarui kredensial pada setiap instance saat memutar kredensial Anda. AWS

Note

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 tidak membuat kebijakan kepercayaan, Anda akan menemui kesalahan `AccessDenied`. Untuk informasi selengkapnya, lihat [Memodifikasi kebijakan kepercayaan peran](#) dalam Panduan Pengguna IAM.

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 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.

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 disebarluaskan ke semua instans.

Note

Kredensial peran Amazon EC2 IAM tidak tunduk pada durasi sesi maksimum yang dikonfigurasi dalam peran. Untuk informasi selengkapnya, lihat [Menggunakan IAM role](#) 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 komunikasi Windows-ke-Windows, gunakan grup dan peran Windows yang sudah 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](#)
- [Mengambil kredensial keamanan dari metadata instans](#)
- [Berikan izin kepada pengguna untuk meneruskan peran IAM ke instans](#)
- [Cara menggunakan IAM role](#)

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 API AWS CLI, 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 melampirkan peran IAM ke instance, tentukan nama profil instance.

profil instans hanya dapat berisi satu IAM role saja. Batas ini tidak dapat dinaikkan.

Untuk informasi selengkapnya, lihat [Profil Instans](#) dalam 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.

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-jenis layanan yang dapat mengekspos kredensial Anda termasuk proksi HTTP, layanan-layanan validator HTML/CSS, dan prosesor XML yang mendukung inklusi XML.

Perintah berikut akan mengambil kredensial keamanan untuk IAM role yang mempunyai nama `s3access`.

Linux

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/iam/security-credentials/s3access
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

Windows

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/meta-data/iam/security-credentials/s3access
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

Berikut ini adalah output contoh.

```
{
  "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 Tools untuk PowerShell perintah Windows yang berjalan pada instance, Anda tidak perlu secara eksplisit mendapatkan kredensial keamanan sementara — AWS SDK, AWS CLI, dan Tools untuk Windows PowerShell secara otomatis mendapatkan kredensial dari layanan metadata instans 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 AWS Sumber Daya](#) di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang metadata instans, lihat [Cara menggunakan metadata instans](#). Untuk informasi tentang alamat IP metadata instans, lihat [Mengambil metadata instans](#).

Berikan izin kepada pengguna untuk meneruskan peran IAM ke instans

Guna memungkinkan pengguna untuk meluncurkan instans dengan peran IAM atau untuk melampirkan atau mengganti peran IAM untuk instans yang sudah ada, Anda harus memberikan izin kepada pengguna untuk menggunakan tindakan API berikut:

- iam:PassRole
- ec2:AssociateIamInstanceProfile
- ec2:ReplaceIamInstanceProfileAssociation

Sebagai contoh, kebijakan IAM berikut ini akan memberikan izin kepada para pengguna untuk meluncurkan instans dengan IAM role, atau untuk melampirkan atau mengganti IAM role untuk instans yang sudah ada menggunakan AWS CLI.

Note

Jika Anda ingin kebijakan tersebut memberikan akses kepada pengguna ke semua peran Anda, tentukan sumber daya sebagai * dalam kebijakan tersebut. Namun demikian, harap mempertimbangkan prinsip [hak akses paling rendah](#) sebagai praktik terbaik.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RunInstances",
        "ec2:AssociateIamInstanceProfile",
        "ec2:ReplaceIamInstanceProfileAssociation"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/DevTeam*"
    }
  ]
}
```

Untuk memberikan izin kepada para pengguna untuk meluncurkan instans dengan IAM role, atau untuk melampirkan atau mengganti IAM role untuk instans yang sudah ada menggunakan konsol Amazon EC2, Anda harus memberikan mereka izin untuk menggunakan `iam:ListInstanceProfiles`, `iam:PassRole`, `ec2:AssociateIamInstanceProfile`, dan `ec2:ReplaceIamInstanceProfileAssociation` selain izin lain yang mungkin mereka butuhkan. Untuk kebijakan-kebijakan contoh, lihat [Kebijakan contoh yang digunakan dalam konsol Amazon EC2](#).

Cara menggunakan IAM role

Anda dapat membuat IAM role dan melampirkan peran itu ke instans selama atau setelah peluncuran dilakukan. Anda juga dapat mengganti atau melepaskan IAM role untuk instans.

Daftar Isi

- [Membuat IAM role](#)
- [Meluncurkan instans dengan IAM role](#)
- [Melampirkan IAM role ke instans](#)
- [Mengganti IAM role](#)
- [Melepaskan IAM role terlampir](#)

- [Membuat kebijakan untuk IAM role Anda berdasarkan aktivitas akses](#)

Membuat IAM role

Anda harus membuat IAM role sebelum Anda dapat meluncurkan instans dengan peran tersebut atau melampirkannya pada instans.

Console

Untuk membuat IAM role menggunakan konsol IAM

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Dalam panel navigasi, pilih Peran, lalu Buat peran.
3. Pada halaman Pilih entitas tepercaya, pilih Layanan AWS, lalu pilih kasus penggunaan EC2. Pilih Berikutnya.
4. Pada halaman Tambahkan izin, pilih kebijakan yang memberi instans Anda akses ke sumber daya yang dibutuhkan. Pilih Berikutnya.
5. Di halaman Nama, tinjau, dan buat, masukkan nama dan deskripsi untuk peran tersebut. Secara opsional, tambahkan tanda ke peran. Pilih Buat peran.

Command line

Contoh berikut membuat IAM role dengan kebijakan yang memungkinkan peran tersebut untuk menggunakan bucket Amazon Simple Storage Service (Amazon S3).

Untuk membuat IAM role dan profil instans (AWS CLI)

1. Buat kebijakan kepercayaan berikut dan simpan di file teks dengan nama `ec2-role-trust-policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": { "Service": "ec2.amazonaws.com" },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

```
}
```

2. Buat peran `s3access` dan tentukan kebijakan kepercayaan yang Anda buat menggunakan perintah `create-role`.

```
aws iam create-role \  
  --role-name s3access \  
  --assume-role-policy-document file://ec2-role-trust-policy.json
```

Contoh tanggapan

```
{  
  "Role": {  
    "AssumeRolePolicyDocument": {  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Action": "sts:AssumeRole",  
          "Effect": "Allow",  
          "Principal": {  
            "Service": "ec2.amazonaws.com"  
          }  
        }  
      ]  
    },  
    "RoleId": "AROAIIZKPBKS2LEXAMPLE",  
    "CreateDate": "2013-12-12T23:46:37.247Z",  
    "RoleName": "s3access",  
    "Path": "/",  
    "Arn": "arn:aws:iam::123456789012:role/s3access"  
  }  
}
```

3. Buat kebijakan akses dan simpan kebijakan tersebut di file teks dengan nama `ec2-role-access-policy.json`. Sebagai contoh, kebijakan ini akan memberikan izin administratif untuk Amazon Simple Storage Service (Amazon S3) untuk aplikasi-aplikasi yang berjalan pada instans.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {
```

```

    "Effect": "Allow",
    "Action": ["s3:*"],
    "Resource": ["*"]
  }
]
}

```

- Lampirkan kebijakan akses ke peran menggunakan perintah [put-role-policy](#).

```

aws iam put-role-policy \
  --role-name s3access \
  --policy-name S3-Permissions \
  --policy-document file://ec2-role-access-policy.json

```

- Buat profil instans dengan nama s3access-profile menggunakan perintah [create-instance-profile](#).

```

aws iam create-instance-profile --instance-profile-name s3access-profile

```

Contoh tanggapan

```

{
  "InstanceProfile": {
    "InstanceProfileId": "AIPAJTLPJLEGREXAMPLE",
    "Roles": [],
    "CreateDate": "2013-12-12T23:53:34.093Z",
    "InstanceProfileName": "s3access-profile",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:instance-profile/s3access-profile"
  }
}

```

- Tambahkan peran s3access pada profil instans s3access-profile.

```

aws iam add-role-to-instance-profile \
  --instance-profile-name s3access-profile \
  --role-name s3access

```

Atau, Anda dapat menggunakan AWS Tools for Windows PowerShell perintah berikut:

- [New-IAMRole](#)

- [Daftarkan-IAM RolePolicy](#)
- [IAM baru InstanceProfile](#)

Meluncurkan instans dengan IAM role

Setelah Anda membuat IAM role, Anda dapat meluncurkan instans, dan mengaitkan peran tersebut pada instans saat dilakukan peluncuran.

 Important

Setelah Anda membuat IAM role, mungkin dibutuhkan beberapa detik bagi izin tersebut untuk tersebar. Jika upaya pertama Anda untuk meluncurkan instans dengan peran gagal, tunggu beberapa detik sebelum Anda mencobanya kembali. Untuk informasi selengkapnya, lihat [Pemecahan Masalah peran IAM](#) dalam Panduan Pengguna IAM.

New console

Untuk meluncurkan instans dengan IAM role (konsol)

1. Ikuti prosedur untuk [meluncurkan instans](#).
2. Perluas Detail lanjutan, dan untuk profil instans IAM, pilih peran IAM yang Anda buat.

 Note

Daftar Profil instans IAM akan menampilkan nama profil instans yang telah Anda buat saat membuat peran IAM. Jika Anda membuat IAM role menggunakan konsol, maka profil instans dibuat untuk Anda dan diberi nama yang sama dengan peran tersebut. Jika Anda membuat peran IAM menggunakan API AWS CLI, atau AWS SDK, Anda mungkin telah menamai profil instans Anda secara berbeda.

3. Konfigurasi detail lain yang Anda perlukan untuk instans atau terima defaultnya, dan pilih pasangan kunci. Untuk informasi tentang bidang di wizard peluncuran instans, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
4. Di panel Ringkasan, tinjau konfigurasi instans Anda, lalu pilih Luncurkan instans.
5. Jika Anda menggunakan tindakan Amazon EC2 API dalam aplikasi Anda, ambil kredensial AWS keamanan yang tersedia pada instans dan gunakan untuk menandatangani permintaan. AWS SDK melakukan ini untuk Anda.

IMDSv2

Untuk instance Linux, lihat contoh berikut:

```
[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/iam/security-credentials/role_name
```

Untuk instance Windows, lihat contoh berikut:

```
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/iam/security-  
credentials/role_name
```

IMDSv1

Untuk instance Linux, lihat contoh berikut:

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-  
credentials/role_name
```

Untuk instance Windows, lihat contoh berikut:

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/  
security-credentials/role_name
```

Old console

Untuk meluncurkan instans dengan IAM role (konsol)

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari dasbor, pilih Luncurkan instans.
3. Pilih AMI dan tipe instans dan kemudian pilih Berikutnya: Konfigurasi Detail Instans.

4. Pada halaman Konfigurasi Detail Instans, untuk IAM role (peran IAM), pilih IAM role yang sudah Anda ciptakan.

 Note

Daftar IAM role akan menampilkan nama profil instans yang telah Anda buat saat Anda membuat IAM role Anda. Jika Anda membuat IAM role menggunakan konsol, maka profil instans dibuat untuk Anda dan diberi nama yang sama dengan peran tersebut. Jika Anda membuat peran IAM menggunakan API AWS CLI, atau AWS SDK, Anda mungkin telah menamai profil instans Anda secara berbeda.

5. Lakukan konfigurasi pada detail-detail lainnya, lalu ikuti petunjuk melalui bagian lain pemandu, atau pilih Review and Launch (Tinjau dan Luncurkan) untuk menerima pengaturan default dan langsung buka halaman Tinjau Peluncuran Instans.
6. Tinjau pengaturan Anda, lalu pilih Launch (Peluncuran) untuk memilih pasangan kunci dan meluncurkan instans Anda.
7. Jika Anda menggunakan tindakan Amazon EC2 API dalam aplikasi Anda, ambil kredensial AWS keamanan yang tersedia pada instans dan gunakan untuk menandatangani permintaan. AWS SDK melakukan ini untuk Anda.

IMDSv2

Untuk instance Linux, lihat contoh berikut:

```
[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/iam/security-credentials/role_name
```

Untuk instance Windows, lihat contoh berikut:

```
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/iam/security-
  credentials/role_name
```

IMDSv1

Untuk instance Linux, lihat contoh berikut:

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

Untuk instance Windows, lihat contoh berikut:

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

Command line

Anda dapat menggunakan AWS CLI untuk mengaitkan peran dengan instance selama peluncuran. Anda harus menentukan profil instans dalam perintah.

Untuk meluncurkan instans dengan IAM role (AWS CLI)

1. Gunakan perintah [run-instances](#) untuk meluncurkan instans menggunakan profil instans. Contoh berikut menunjukkan cara meluncurkan instans dengan profil instans.

```
aws ec2 run-instances \  
  --image-id ami-11aa22bb \  
  --iam-instance-profile Name="s3access-profile" \  
  --key-name my-key-pair \  
  --security-groups my-security-group \  
  --subnet-id subnet-1a2b3c4d
```

Atau, gunakan PowerShell perintah [New-EC2InstanceTools](#) for Windows.

2. Jika Anda menggunakan tindakan Amazon EC2 API dalam aplikasi Anda, ambil kredensial AWS keamanan yang tersedia pada instans dan gunakan untuk menandatangani permintaan. AWS SDK melakukan ini untuk Anda.

```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

Melampirkan IAM role ke instans

Untuk melampirkan IAM role pada instans yang tidak memiliki peran, instans tersebut dapat berada dalam status `stopped` atau `running`.

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, pilih Actions (Tindakan), Security (Keamanan), Modify IAM role (Modifikasi IAM role).
4. Pilih IAM role yang akan dilampirkan ke instans Anda, lalu pilih Save (Simpan).

Command line

Untuk melampirkan IAM role ke instans (AWS CLI)

1. Jika perlu, deskripsikan instans Anda untuk mendapatkan ID instans yang akan digunakan untuk melampirkan peran tersebut padanya.

```
aws ec2 describe-instances
```

2. Gunakan perintah [associate-iam-instance-profile](#) untuk melampirkan peran IAM ke instans dengan menentukan profil instans. Anda dapat menggunakan Amazon Resource Name (ARN) dari profil instans tersebut, atau Anda dapat menggunakan namanya.

```
aws ec2 associate-iam-instance-profile \  
  --instance-id i-1234567890abcdef0 \  
  --iam-instance-profile Name="TestRole-1"
```

Contoh tanggapan

```
{  
  "IamInstanceProfileAssociation": {  
    "InstanceId": "i-1234567890abcdef0",  
    "State": "associating",  
    "AssociationId": "iip-assoc-0dbd8529a48294120",  
    "IamInstanceProfile": {
```

```
    "Id": "AIPAJLNLDX3AMYZNWYYAY",  
    "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-1"  
  }  
}  
}
```

Atau, gunakan alat berikut untuk PowerShell perintah Windows:

- [Get-EC2Instance](#)
- [Register-EC2IamInstanceProfile](#)

Mengganti IAM role

Untuk mengganti IAM role pada instans yang sudah memiliki IAM role yang dilampirkan padanya, instans tersebut harus berada dalam status `running`. 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 IAM role untuk instans

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, pilih Actions (Tindakan), Security (Keamanan), Modify IAM role (Modifikasi IAM role).
4. Pilih IAM role yang akan dilampirkan ke instans Anda, lalu pilih Save (Simpan).

Command line

Untuk mengganti IAM role untuk instans (AWS CLI)

1. Jika perlu, deskripsikan asosiasi profil instans IAM Anda untuk mendapatkan ID asosiasi yang akan digantikan profil instans IAM tersebut.

```
aws ec2 describe-iam-instance-profile-associations
```

- Gunakan perintah [replace-iam-instance-profile-association](#) untuk mengganti profil instans IAM dengan menentukan ID asosiasi untuk profil instans yang sudah ada tersebut, dan juga ARN atau nama dari profil instans yang seharusnya menggantikannya.

```
aws ec2 replace-iam-instance-profile-association \  
  --association-id iip-assoc-0044d817db6c0a4ba \  
  --iam-instance-profile Name="TestRole-2"
```

Contoh tanggapan

```
{  
  "IamInstanceProfileAssociation": {  
    "InstanceId": "i-087711ddaf98f9489",  
    "State": "associating",  
    "AssociationId": "iip-assoc-09654be48e33b91e0",  
    "IamInstanceProfile": {  
      "Id": "AIPAJCJEDKX7QYHWYK7GS",  
      "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"  
    }  
  }  
}
```

Atau, gunakan alat berikut untuk PowerShell perintah Windows:

- [Get-EC2IamInstanceProfileAssociation](#)
- [Set-EC2IamInstanceProfileAssociation](#)

Melepaskan IAM role terlampir

Anda dapat melepaskan IAM role terlampir dari instans yang berjalan atau dihentikan.

Console

Untuk melepaskan IAM role terlampir dari instans

- Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
- Di panel navigasi, pilih Instans.
- Pilih instans, pilih Actions (Tindakan), Security (Keamanan), Modify IAM role (Modifikasi IAM role).

4. Untuk IAM role, pilih No IAM Role (Tanpa IAM Role). Pilih Save (Simpan).
5. Dalam kotak dialog konfirmasi, masukkan Detach (Lepaskan), lalu pilih Detach (Lepaskan).

Command line

Untuk melepaskan IAM role terlampir dari instans (AWS CLI)

1. Jika perlu, gunakan [describe-iam-instans-profile-associations](#) untuk mendeskripsikan asosiasi profil instans IAM Anda dan untuk mendapatkan ID asosiasi yang akan dilepaskan profil instans IAM.

```
aws ec2 describe-iam-instance-profile-associations
```

Contoh tanggapan

```
{
  "IamInstanceProfileAssociations": [
    {
      "InstanceId": "i-088ce778fbfeb4361",
      "State": "associated",
      "AssociationId": "iip-assoc-0044d817db6c0a4ba",
      "IamInstanceProfile": {
        "Id": "AIPAJEDNCAA64SSD265D6",
        "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
      }
    }
  ]
}
```

2. Gunakan perintah [disassociate-iam-instance-profile](#) untuk melepaskan profil instans IAM menggunakan ID asosiasi dari profil instans tersebut.

```
aws ec2 disassociate-iam-instance-profile --association-id iip-  
assoc-0044d817db6c0a4ba
```

Contoh tanggapan

```
{
  "IamInstanceProfileAssociation": {
    "InstanceId": "i-087711ddaf98f9489",
```

```
"State": "disassociating",
"AssociationId": "iip-assoc-0044d817db6c0a4ba",
"IamInstanceProfile": {
  "Id": "AIPAJEDNCAA64SSD265D6",
  "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
}
}
```

Atau, gunakan alat berikut untuk PowerShell perintah Windows:

- [Get-EC2IamInstanceProfileAssociation](#)
- [Unregister-EC2IamInstanceProfile](#)

Membuat kebijakan untuk IAM role Anda berdasarkan aktivitas akses

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 menghasilkan templat kebijakan yang berisi izin yang telah digunakan oleh peran dalam rentang tanggal yang ditentukan. 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 diperlukan 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 mempelajari selengkapnya, lihat [Membuat kebijakan berdasarkan aktivitas akses](#) di Panduan Pengguna IAM.

Mengakses Amazon EC2 menggunakan titik akhir VPC antarmuka

Anda dapat meningkatkan postur keamanan VPC Anda dengan membuat koneksi privat antara VPC dan Amazon EC2. Anda dapat mengakses Amazon EC2 seolah-olah berada di VPC Anda, tanpa menggunakan gateway internet, perangkat NAT, koneksi VPN, atau koneksi. AWS Direct Connect Instans dalam VPC Anda tidak memerlukan alamat IP publik untuk mengakses Amazon EC2.

Untuk informasi selengkapnya, lihat [Akses Layanan AWS melalui AWS PrivateLink](#) di 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-tindakan Amazon EC2 API.

Untuk informasi selengkapnya, lihat [Mengakses titik akhir VPC antarmuka Layanan AWS menggunakan antarmuka di Panduan.AWS PrivateLink](#)

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

Jika kebijakan non-default diterapkan ke titik akhir VPC antarmuka untuk Amazon EC2, permintaan API tertentu yang gagal, seperti yang gagalRequestLimitExceeded, mungkin tidak dicatat atau Amazon. AWS CloudTrail 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.

```
{
  "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"
        }
      }
    }
  ]
}
```

Manajemen pembaruan untuk instans Windows 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 mengotomatiskan 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 - [Memperbarui AL2023 di Panduan Pengguna Amazon Linux 2023](#).
- AL2 — [Kelola perangkat lunak pada instans Amazon Linux 2 Anda](#) di Panduan Pengguna Amazon Linux 2.
- Contoh Windows — [the section called “Manajemen pembaruan”](#).

Praktik terbaik keamanan untuk instans Windows

Kami menyarankan Anda mengikuti praktik terbaik keamanan ini untuk instans Windows Anda.

Daftar Isi

- [Praktik terbaik keamanan tingkat tinggi](#)
- [Manajemen pembaruan](#)
- [Manajemen konfigurasi](#)
- [Manajemen perubahan](#)
- [Audit dan akuntabilitas untuk instans Windows Amazon EC2](#)

Praktik terbaik keamanan tingkat tinggi

Anda harus mematuhi praktik terbaik keamanan tingkat tinggi berikut untuk instance Windows Anda:

- Akses paling sedikit — Berikan akses hanya ke sistem dan lokasi yang dipercaya 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 subnet publik/pribadi Amazon VPC untuk melapisi keamanan di beberapa lokasi dalam arsitektur. Dalam instance Windows, pelanggan dapat menggunakan Windows Firewall untuk lebih lanjut melapisi defense-in-depth strategi dalam penyebaran mereka. Cukup instal komponen dan aplikasi OS yang diperlukan agar sistem berfungsi sebagaimana peruntukannya. Konfigurasi layanan infrastruktur, seperti IIS, untuk dijalankan di bawah akun layanan atau untuk menggunakan fitur, seperti identitas kolom aplikasi, agar dapat mengakses sumber daya secara lokal dan jarak jauh di seluruh infrastruktur Anda.

- **Keistimewaan terkecil** — Tentukan kumpulan hak istimewa minimum yang dibutuhkan instance 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 up-to-date keamanan dan suite perlindungan berbasis host yang mencakup anti-virus, anti-malware, deteksi/pencegahan intrusi, 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. Juga, manfaatkan Just Enough Administration (JEA) dengan Windows PowerShell DSC untuk membatasi akses administratif ke fungsi minimum yang diperlukan.
- **Manajemen Patch** — Menerapkan proses yang secara teratur menambal, memperbarui, dan mengamankan sistem operasi dan aplikasi pada instans EC2 Anda.
- **Log audit** — Akses audit dan semua perubahan pada instans Amazon EC2 untuk memverifikasi integritas server dan memastikan hanya perubahan resmi yang dibuat. Manfaatkan fitur seperti [Enhanced Logging for IIS](#) untuk meningkatkan kemampuan logging default. AWS kemampuan seperti VPC Flow Logs dan juga AWS CloudTrail tersedia untuk mengaudit akses jaringan, termasuk permintaan yang diizinkan/ditolak dan panggilan API, masing-masing.

Manajemen pembaruan

Untuk memastikan hasil terbaik saat menjalankan Windows Server di Amazon EC2, sebaiknya Anda menerapkan praktik terbaik berikut:

- [Configure Windows Update](#)
- [Update drivers](#)
- [Use the latest Windows AMIs](#)
- [Test performance before migration](#)
- [Update launch agents](#)
- Reboot instance Windows Anda setelah Anda menginstal pembaruan. Untuk informasi selengkapnya, lihat [Menyalakan ulang instans Anda](#).

Untuk informasi selengkapnya, tentang cara meningkatkan atau migrasi instans Windows ke versi Windows Server yang lebih baru, lihat [Mutakhirkan instans Amazon EC2 Windows ke versi Windows Server yang lebih baru](#).

Konfigurasi Pembaruan Windows

Secara default, instance 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. Bergantung 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- UpgradeWindows AWSDrivers](#) 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 [riwayat versi AWS Windows AMI](#).
- 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 penerapan kepatuhan, lihat [AMI Windows Khusus](#) di Referensi AWS AMI Windows.

Menguji performa sistem/aplikasi 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.

Memperbarui agen peluncuran

Perbarui ke agen EC2Launch v2 terkini untuk memastikan bahwa peningkatan terkini diterapkan di seluruh armada Anda. Untuk informasi selengkapnya, lihat [the section called "Migrasi"](#).

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	✓	✓

Versi Windows	EC2Launch v1	EC2Launch v2
2022		✓

- Untuk informasi selengkapnya tentang memperbarui ke EC2launch v2, lihat. [the section called “Pasang”](#)
- 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 khusus untuk memastikan bahwa semua instance Windows diluncurkan dengan kontrol keamanan standar. Garis dasar keamanan dapat dimasukkan ke dalam AMI, di-bootstrap secara dinamis saat instans EC2 diluncurkan, atau dikemas sebagai produk untuk distribusi seragam melalui portofolio Service Catalog. 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 mewaspadaai 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 diterapkan 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, AWS CloudFormation seperti, sebaiknya gunakan fitur OS asli, seperti [Microsoft PowerShell Windows](#) 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 patching aplikasi, upgrade perangkat lunak, atau pembaruan file konfigurasi).

AWS menyediakan beberapa alat untuk membantu mengelola perubahan pada AWS sumber daya, termasuk AWS CloudTrail,, AWS Config AWS CloudFormation, dan AWS Elastic Beanstalk, AWS OpsWorks, dan paket manajemen untuk Manajer Operasi Pusat Sistem dan Manajer Mesin Virtual Pusat Sistem. Perhatikan bahwa Microsoft merilis patch Windows setiap hari Selasa (kadang-kadang bahkan setiap hari) dan AWS memperbarui semua AMI Windows yang dikelola oleh AWS dalam waktu lima hari setelah Microsoft merilis patch. Oleh karena itu, penting untuk terus menambal semua AMI baseline, memperbarui AWS CloudFormation template dan konfigurasi grup Auto Scaling dengan ID AMI terbaru, dan menerapkan alat untuk mengotomatiskan manajemen patch instance 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 skrip dan berulang. 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 audit dan pelacakan 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.

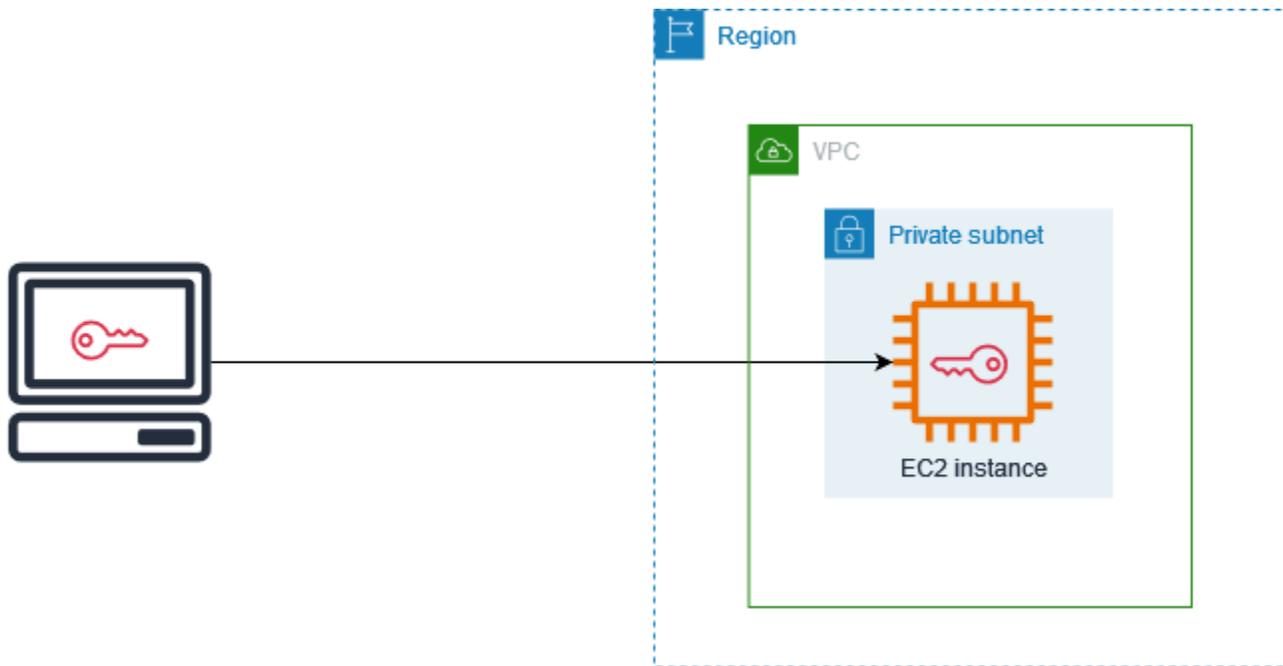
Selain alat manajemen sistem Microsoft, pelanggan dapat menggunakan Amazon CloudWatch untuk memantau pemanfaatan CPU instans, kinerja disk, I/O jaringan, dan melakukan pemeriksaan status host dan instance. Agen peluncuran EC2config, EC2launch, dan EC2launch v2 menyediakan akses ke fitur-fitur tambahan dan canggih untuk instance 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 instans Anda. Untuk instance Windows, kunci pribadi diperlukan untuk mendekripsi kata sandi administrator, yang kemudian Anda gunakan untuk terhubung 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 instance Anda yang menggunakan key pair.



Ketika Anda meluncurkan sebuah instance, Anda dapat [menentukan key pair](#), sehingga Anda dapat terhubung ke instance Anda menggunakan metode yang memerlukan key pair. 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, ketika instance Anda melakukan booting untuk pertama kalinya, kunci publik yang Anda tentukan saat peluncuran ditempatkan pada instance Linux Anda dalam entri di dalamnya `~/.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, lihat [Connect 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 pribadinya Anda hilangkan. Untuk informasi selengkapnya, lihat [Saya kehilangan kunci privat. Bagaimana caranya terhubung ke instans Linux saya?](#).

Sebagai alternatif dari pasangan kunci, Anda dapat menggunakan [AWS Systems Manager Session Manager](#) untuk terhubung ke instance Anda dengan shell berbasis browser satu-klik interaktif atau (`ssh`).
AWS Command Line Interface AWS CLI

Daftar Isi

- [Membuat key pair untuk instans Amazon EC2](#)
- [Menandai key pair](#)
- [Jelaskan pasangan kunci Anda](#)
- [Menghapus pasangan kunci Anda](#)
- [Menambahkan atau menghapus kunci publik pada instance Linux Anda](#)
- [Lakukan verifikasi terhadap sidik jari pasangan kunci Anda](#)

Membuat key pair untuk instans Amazon EC2

Anda dapat menggunakan Amazon EC2 untuk membuat pasangan kunci, atau Anda dapat menggunakan alat pihak ketiga untuk membuat pasangan kunci, lalu mengimpornya ke Amazon EC2.

Amazon EC2 mendukung kunci RSA SSH-2 2048-bit untuk instance Linux dan Windows. Amazon EC2 juga mendukung kunci ED25519 untuk instans Linux.

Untuk langkah-langkah untuk terhubung ke instance Linux Anda menggunakan SSH setelah Anda membuat key pair, lihat [the section called “Hubungkan ke instans Linux Anda”](#).

Untuk langkah-langkah untuk terhubung ke instance Windows Anda menggunakan RDP setelah Anda membuat key pair, lihat [the section called “Hubungkan ke instans Windows Anda”](#)

Daftar Isi

- [Membuat pasangan kunci menggunakan Amazon EC2](#)
- [Buat key pair 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 AWS Support 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 key pair yang sesuai untuk sistem operasi Anda:

(Instance Linux) Untuk tipe pasangan Kunci, pilih RSA atau ED25519.

(Instans Windows) Untuk jenis pasangan kunci, pilih RSA. Kunci ED25519 tidak didukung untuk instance 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. (Instance Linux) Jika Anda berencana untuk menggunakan klien SSH di komputer macOS atau Linux untuk terhubung ke instance Linux Anda, gunakan perintah berikut untuk mengatur izin file kunci pribadi 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`.

Untuk `--key-name`, tentukan nama untuk kunci publik. Nama dapat terdiri dari hingga 255 karakter ASCII.

Untuk `--key-type`, tentukan salah satu, `rsa` atau `ed25519`. Jika Anda tidak menyertakan parameter `--key-type`, kunci `rsa` akan dibuat secara default. Perhatikan bahwa kunci ED25519 tidak didukung untuk instans Windows.

Untuk `--key-format`, tentukan salah satu, `pem` atau `ppk`. Jika Anda tidak menyertakan parameter `--key-format`, file `pem` akan dibuat secara default.

`--query "KeyMaterial"` mencetak materi kunci privat ke output.

`--output text > my-key-pair.pem` menyimpan materi kunci privat di file dengan ekstensi yang ditentukan. Ekstensi dapat berupa `.pem` atau `.ppk`. Kunci privat dapat memiliki nama yang 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
```

```
--output text > my-key-pair.pem
```

- (Instance Linux) Jika Anda berencana untuk menggunakan klien SSH di komputer macOS atau Linux untuk terhubung ke instance Linux Anda, gunakan perintah berikut untuk mengatur izin file kunci pribadi 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](#) AWS Tools for Windows PowerShell perintah sebagai berikut untuk menghasilkan kunci dan menyimpannya ke .ppk file .pem atau.

Untuk -KeyName, tentukan nama untuk kunci publik. Nama dapat terdiri dari hingga 255 karakter ASCII.

Untuk -KeyType, tentukan salah satu, `rsa` atau `ed25519`. Jika Anda tidak menyertakan parameter -KeyType, kunci `rsa` akan dibuat secara default. Perhatikan bahwa kunci ED25519 tidak didukung untuk instans Windows.

Untuk -KeyFormat, tentukan salah satu, `pem` atau `ppk`. Jika Anda tidak menyertakan parameter -KeyFormat, file `pem` akan dibuat secara default.

KeyMaterial mencetak materi kunci privat ke output.

Out-File -Encoding `ascii` -FilePath `C:\path\my-key-pair.pem` menyimpan materi kunci privat di file dengan ekstensi yang ditentukan. Ekstensinya bisa `.pem` atau `.ppk`. Kunci privat dapat memiliki nama yang berbeda dari nama kunci publik, tetapi untuk kemudahan penggunaan, gunakan nama yang sama.

```
PS C:\> (New-EC2KeyPair -KeyName "my-key-pair" -KeyType "rsa" -KeyFormat "pem").KeyMaterial | Out-File -Encoding ascii -FilePath C:\path\my-key-pair.pem
```

Buat key pair menggunakan AWS CloudFormation

Saat Anda membuat key pair baru menggunakan AWS CloudFormation, kunci pribadi disimpan ke AWS Systems Manager Parameter Store. 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 key pair menggunakan AWS CloudFormation

1. Tentukan [AWS::EC2::KeyPair](#) sumber daya 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

AWS CloudFormation Untuk mengaktifkan mengelola parameter Parameter Store atas nama Anda, peran IAM yang diambil oleh AWS 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 AWS CloudFormation.

Ketika AWS CloudFormation menghapus key pair yang dibuat atau diimpor oleh stack, ia melakukan pemeriksaan izin untuk menentukan apakah Anda memiliki izin untuk menghapus parameter, meskipun AWS CloudFormation membuat parameter hanya ketika membuat key pair, bukan ketika mengimpor key pair. AWS CloudFormation tes untuk izin yang diperlukan menggunakan nama parameter fabrikasi 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

Instans Linux

Alih-alih menggunakan Amazon EC2 untuk membuat pasangan kunci, Anda dapat membuat pasangan kunci RSA atau ED25519 menggunakan alat dari pihak ketiga, dan kemudian mengimpor kunci publik ke Amazon EC2.

Ketentuan untuk pasangan kunci

- Tipe yang didukung: RSA dan ED25519. Amazon EC2 tidak menerima kunci DSA.
- Format yang didukung:
 - Format kunci publik OpenSSH (format dalam). `~/.ssh/authorized_keys` Jika Anda terhubung menggunakan SSH saat menggunakan API Connect Instans EC2, format SSH2 juga didukung.
 - Format file kunci privat SSH harus dalam format PEM atau PPK
 - (Khusus RSA) Format DER dengan encode Base64
 - (Khusus RSA) Format file kunci publik SSH sebagaimana yang ditentukan dalam [RFC 4716](#)
- Panjang yang didukung: 1024, 2048, dan 4096. Jika Anda terhubung menggunakan SSH saat menggunakan API Connect Instans 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, Anda bisa menggunakan Java, Ruby, Python, dan banyak bahasa pemrograman lainnya yang menyediakan pustaka standar yang dapat Anda gunakan untuk membuat pasangan kunci RSA atau ED25519.

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. Sebagai contoh, ~/.ssh/my-key-pair.pub. 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.

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.

Instans Windows

Alih-alih menggunakan Amazon EC2 untuk membuat pasangan kunci Anda, Anda dapat membuat pasangan kunci RSA menggunakan alat dari pihak ketiga, dan kemudian mengimpor kunci publik ke Amazon EC2.

Ketentuan untuk pasangan kunci

- Tipe yang didukung: RSA. Amazon EC2 tidak menerima kunci DSA.

Note

Kunci ED25519 tidak didukung untuk instans Windows.

- Format yang didukung:
 - Format kunci publik OpenSSH
 - Format file kunci privat SSH harus dalam format PEM atau PPK
 - (Khusus RSA) Format DER dengan encode Base64
 - (Khusus RSA) Format file kunci publik SSH sebagaimana yang ditentukan dalam [RFC 4716](#)
- Panjang yang didukung: 1024, 2048, dan 4096.

Cara membuat pasangan kunci menggunakan alat pihak ketiga

1. Buat pasangan kunci dengan alat pihak ketiga yang Anda kehendaki. Misalnya, Anda dapat menggunakan ssh-keygen (alat yang disediakan dengan instalasi OpenSSH standar). Atau, Java, Ruby, Python, dan banyak bahasa pemrograman lainnya menyediakan perpustakaan standar yang dapat Anda gunakan untuk menciptakan sebuah key pair RSA.

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. Sebagai contoh, C:\keys\my-key-pair.pub. Ekstensi nama file untuk file ini bukan hal penting.
3. Simpan kunci privat ke file lokal yang memiliki ekstensi .pem atau .ppk. Misalnya, C:\keys\my-key-pair.pem atau C:\keys\my-key-pair.ppk. Ekstensi nama file untuk file ini penting karena hanya .pem file yang dapat dipilih saat menghubungkan ke instance Windows Anda dari konsol EC2.

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 CLI .

Cara melakukan verifikasi terhadap pasangan kunci yang berhasil diimpor

Gunakan perintah [describe-key-pairs](#) AWS CLI .

PowerShell

Cara mengimpor kunci publik ke Amazon EC2

Gunakan perintah [Import-EC2KeyPair](#) AWS Tools for Windows PowerShell .

Cara melakukan verifikasi terhadap pasangan kunci yang berhasil diimpor

Gunakan perintah [Get-EC2KeyPair](#) AWS Tools for Windows PowerShell .

Menandai key pair

Untuk membantu mengkategorikan dan mengelola pasangan kunci yang telah dibuat menggunakan Amazon EC2 atau diimpor ke Amazon EC2, Anda dapat menandainya dengan metadata khusus. Untuk informasi selengkapnya tentang cara memberikan tanda, lihat [Tandai sumber daya Amazon EC2 Anda](#).

Console

Untuk melihat, menambah, atau menghapus tag untuk key pair

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Pasangan Kunci.
3. Pilihlah kunci publik, dan kemudian pilih Tindakan, Kelola tanda.
4. Halaman Kelola tanda akan menampilkan tanda-tanda yang ditetapkan untuk kunci publik tersebut.
 - Untuk menambahkan tanda, pilih Tambahkan tanda, dan masukkan kunci dan nilai tanda. Anda dapat menambahkan hingga 50 tanda untuk setiap kunci. Untuk informasi selengkapnya, lihat [Pembatasan tanda](#).
 - Untuk menghapus tanda, pilih Remove (Hapus) yang ada di samping tanda yang akan dihapus.
5. Pilih Save (Simpan).

AWS CLI

Untuk melihat tag untuk pasangan kunci Anda

Gunakan perintah [describe-tags](#) AWS CLI . Dalam contoh berikut, Anda mendeskripsikan tanda untuk semua kunci publik Anda.

```
aws ec2 describe-tags --filters "Name=resource-type,Values=key-pair"
```

```
{
  "Tags": [
    {
      "Key": "Environment",
      "ResourceId": "key-0123456789EXAMPLE",
```

```

    "ResourceType": "key-pair",
    "Value": "Production"
  },
  {
    "Key": "Environment",
    "ResourceId": "key-9876543210EXAMPLE",
    "ResourceType": "key-pair",
    "Value": "Production"
  }
]
}

```

Untuk mendeskripsikan tag untuk key pair

Gunakan perintah [describe-key-pairs](#) AWS CLI .

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789EXAMPLE
```

```

{
  "KeyPairs": [
    {
      "KeyName": "MyKeyPair",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyPairId": "key-0123456789EXAMPLE",
      "Tags": [
        {
          "Key": "Environment",
          "Value": "Production"
        }
      ]
    }
  ]
}

```

Untuk menandai key pair

Gunakan perintah [create-tags](#) AWS CLI . Dalam contoh berikut, kunci publik ditandai dengan Key=Cost-Center dan Value=CC-123.

```
aws ec2 create-tags --resources key-0123456789EXAMPLE --tags Key=Cost-Center,Value=CC-123
```

Untuk menghapus sebuah tag dari sebuah key pair

Gunakan perintah [delete-tags](#) AWS CLI . Sebagai contoh, lihat [Contoh](#) di Referensi Perintah AWS CLI .

PowerShell

Untuk melihat tag untuk pasangan kunci Anda

Gunakan perintah [Get-EC2Tag](#).

Untuk mendeskripsikan tag untuk key pair

Gunakan perintah [Get-EC2KeyPair](#).

Untuk menandai key pair

Gunakan perintah [New-EC2Tag](#).

Untuk menghapus sebuah tag dari sebuah key pair

Gunakan perintah [Remove-EC2Tag](#).

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.

Topik

- [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.

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk melihat informasi tentang kunci publik Anda.

Console

Cara menampilkan informasi tentang kunci publik

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).

Key pairs (23) [Info](#)

Filter key pairs

<input type="checkbox"/>	Name	Type	Created	Fingerprint	ID
<input type="checkbox"/>		ed25519	2021/08/05 10:06 GMT+2	xeDxC7/IVRZ8mFlzsKidfQ2FcfWig4C3...	key-
<input type="checkbox"/>		rsa	2020/05/13 17:16 GMT+2	ed:71:62:da:a4:d1:f6:47:61:4b:d1:a7:2...	key-

4. Cara melihat tanda dari kunci publik, pilih kotak centang di samping kunci, dan kemudian pilih Actions (Tindakan), Manage tags (Kelola tanda).

AWS CLI

Cara mendeskripsikan kunci publik

Gunakan perintah [describe-key-pairs](#) dan tentukan parameter `--key-names`.

```
aws ec2 describe-key-pairs --key-names key-pair-name
```

Contoh keluaran

```
{
  "KeyPairs": [
    {
      "KeyPairId": "key-0123456789example",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "key-pair-name",
      "KeyType": "rsa",
    }
  ]
}
```

```

        "Tags": [],
        "CreateTime": "2022-04-28T11:37:26.000Z"
    }
]
}

```

Atau, alih-alih `--key-names`, Anda dapat menentukan parameter `--key-pair-ids` untuk mengidentifikasi kunci publik tersebut.

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example
```

Untuk menampilkan materi kunci publik dalam output, Anda harus menentukan parameter `--include-public-key`.

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

Contoh output – Dalam output, bidang `PublicKey` berisi materi kunci publik.

```

{
  "KeyPairs": [
    {
      "KeyPairId": "key-0123456789example",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "key-pair-name",
      "KeyType": "rsa",
      "Tags": [],
      "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIIj7az1DjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
      "CreateTime": "2022-04-28T11:37:26.000Z"
    }
  ]
}

```

Mengambil materi kunci publik

Anda dapat menggunakan berbagai metode untuk mendapatkan akses ke materi kunci publik. Anda dapat mengambil materi kunci publik dari kunci pribadi yang cocok di komputer lokal Anda, dari metadata instance pada instance yang diluncurkan dengan kunci publik, atau dengan menggunakan

perintah. `describe-key-pairs` AWS CLI Untuk instance Linux, materi kunci publik juga dapat diambil dari `authorized_keys` file pada instance.

Gunakan salah satu metode berikut ini untuk mengambil materi kunci publik.

Instans Linux

From the private key

Cara mengambil materi kunci publik dari kunci privat

Pada komputer Linux atau macOS lokal Anda, Anda dapat menggunakan perintah `ssh-keygen` untuk mengambil kunci publik untuk pasangan kunci Anda. Tentukan jalur tempat Anda mengunduh kunci privat Anda (file `.pem`).

```
ssh-keygen -y -f /path_to_key_pair/my-key-pair.pem
```

Perintah tersebut akan mengembalikan kunci publik, sebagaimana yang ditunjukkan dalam contoh berikut.

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V  
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJ0I0iBXr  
lsLnBItnctkiJ7FbtXJMXLvvwJryDUi1BMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ  
qaeJAAHco+CY/5WtUBkrHmFJr6HcXkvJdWpkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb  
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

Jika perintah tersebut gagal, jalankan perintah berikut untuk memastikan bahwa Anda telah mengubah izin pada file pasangan kunci privat Anda sehingga hanya Anda yang dapat melihatnya.

```
chmod 400 key-pair-name.pem
```

From the instance metadata

Anda dapat menggunakan Layanan Metadata Instans Versi 2 atau Layanan Metadata Instans Versi 1 untuk mengambil kunci publik dari metadata instans.

Note

Jika Anda mengubah pasangan kunci yang Anda gunakan untuk terhubung ke instans, Amazon EC2 tidak akan memperbarui metadata instans untuk menunjukkan kunci publik

baru. Metadata instans akan tetap menunjukkan kunci publik untuk pasangan kunci yang Anda tentukan saat Anda meluncurkan instans.

Cara mengambil materi kunci publik dari metadata instans

Gunakan salah satu perintah berikut dari instans Anda.

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/public-keys/0/openssh-key
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

Contoh output

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCLKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706Vhz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJ0I0iBXrlsLnBItnctkiJ7FbtXJMXLvvwJryDUi1BMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3RbBQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

Untuk informasi selengkapnya tentang metadata instans, lihat [Mengambil metadata instans](#).

From the instance

Jika Anda menentukan pasangan kunci saat meluncurkan instans Linux, maka saat instans melakukan booting untuk pertama kalinya, konten dari kunci publik akan ditempatkan pada instans tersebut di entri dalam `~/.ssh/authorized_keys`.

Cara mengambil materi kunci publik dari instans

1. [Terhubung ke instans Anda](#).
2. Di jendela terminal, buka file `authorized_keys` menggunakan editor teks favorit Anda (seperti `vim` atau `nano`).

```
[ec2-user ~]$ nano ~/.ssh/authorized_keys
```

File `authorized_keys` terbuka, menampilkan kunci publik yang diikuti dengan nama pasangan kunci. Berikut ini adalah contoh entri untuk pasangan kunci dengan nama *key-pair-name*.

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4xyyb/wB96xbiFveSFJu0p/d6RJhJ0I0iBXR
lsLnBItnctkiJ7FbtXJMXLvwWJryDUi1BMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPKYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

From `describe-key-pairs`

Cara mengambil materi kunci publik dari perintah **describe-key-pairs** AWS CLI

Gunakan perintah [describe-key-pairs](#) `describe-key-pairs` dan tentukan parameter `--key-names` untuk mengidentifikasi kunci publik. Untuk menampilkan materi kunci publik dalam keluaran, Anda harus menentukan parameter `--include-public-key`.

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

Contoh output – Dalam output, bidang `PublicKey` berisi materi kunci publik.

```
{
  "KeyPairs": [
    {
      "KeyPairId": "key-0123456789example",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "key-pair-name",
      "KeyType": "rsa",
      "Tags": [],
      "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIIj7az1DjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
      "CreateTime": "2022-04-28T11:37:26.000Z"
    }
  ]
}
```

Atau, alih-alih `--key-names`, Anda dapat menentukan parameter `--key-pair-ids` untuk mengidentifikasi kunci publik tersebut.

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-key
```

Instans Windows

From the private key

Cara mengambil materi kunci publik dari kunci privat

Pada komputer Windows lokal Anda, Anda dapat menggunakan PuTTYgen untuk mendapatkan kunci publik untuk pasangan kunci Anda.

Mulai PuTTYgen dan kemudian pilih Load (Beban). 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.

From the instance metadata

Anda dapat menggunakan Layanan Metadata Instans Versi 2 atau Layanan Metadata Instans Versi 1 untuk mengambil kunci publik dari metadata instans.

Note

Jika Anda mengubah pasangan kunci yang Anda gunakan untuk terhubung ke instans, Amazon EC2 tidak akan memperbarui metadata instans untuk menunjukkan kunci publik baru. Metadata instans akan tetap menunjukkan kunci publik untuk pasangan kunci yang Anda tentukan saat Anda meluncurkan instans.

Cara mengambil materi kunci publik dari metadata instans

Gunakan salah satu perintah berikut dari instans Anda.

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/meta-data/public-keys/0/openssh-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
```

Contoh output

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzo0WbkM4yxyb/wB96xbiFveSFJu0p/d6RJhJ0I0iBXr
lsLnBItnctkiJ7FbtXJMXLvvwJryDUilBMTjYtwB+QhYXUM0zce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJR6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb70z1PnW0yN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

Untuk informasi selengkapnya tentang metadata instans, lihat [Mengambil metadata instans](#).

From describe-key-pairs

Cara mengambil materi kunci publik dari perintah **describe-key-pairs** AWS CLI

Gunakan perintah [describe-key-pairs](#) `describe-key-pairs` dan tentukan parameter `--key-names` untuk mengidentifikasi kunci publik. Untuk menampilkan materi kunci publik dalam keluaran, Anda harus menentukan parameter `--include-public-key`.

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

Contoh output – Dalam output, bidang `PublicKey` berisi materi kunci publik.

```
{
  "KeyPairs": [
    {
      "KeyPairId": "key-0123456789example",
      "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
      "KeyName": "key-pair-name",
      "KeyType": "rsa",
      "Tags": [],
      "PublicKey": "ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIIj7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
      "CreateTime": "2022-04-28T11:37:26.000Z"
    }
  ]
}
```

```
    }  
  ]  
}
```

Atau, alih-alih `--key-names`, Anda dapat menentukan parameter `--key-pair-ids` untuk mengidentifikasi kunci publik tersebut.

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-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.

Cara mengidentifikasi kunci publik yang ditentukan saat peluncuran

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 Detail, di bawah Detail Instans, bidang Nama pasangan kunci menampilkan nama kunci publik yang Anda tentukan saat meluncurkan instans.

Note

Nilai dari bidang Nama pasangan kunci tidak akan berubah meskipun Anda mengubah kunci publik pada instans tersebut, atau menambahkan kunci publik.

Menghapus pasangan kunci Anda

Anda dapat menghapus key pair, yang menghapus kunci publik yang disimpan di Amazon EC2. Menghapus key pair 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 CLI .

PowerShell

Cara menghapus kunci publik Anda di Amazon EC2

Gunakan perintah [Remove-EC2KeyPair](#) AWS Tools for Windows PowerShell .

Menambahkan atau menghapus kunci publik pada instance Linux Anda

Jika Anda kehilangan kunci pribadi, Anda kehilangan akses ke setiap instance yang menggunakan key pair. Untuk informasi lebih lanjut tentang menghubungkan ke instance menggunakan key pair 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`.

Anda dapat mengubah pasangan kunci yang Anda gunakan untuk mengakses akun sistem default instans Anda dengan menambahkan kunci publik baru pada instans tersebut, atau dengan mengganti kunci publik (menghapus kunci publik yang ada dan kemudian menambahkan kunci publik yang baru) pada instans. Anda juga dapat menghapus semua kunci publik dari instans. Untuk menambah atau mengganti pasangan kunci, Anda harus dapat terhubung ke instans Anda.

Anda dapat menambahkan atau mengganti key pair 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 key pair untuk pengguna default, seperti `ec2-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](#) menggunakan kunci privat yang sudah ada.
4. Dengan menggunakan editor teks yang Anda kehendaki, buka file `.ssh/authorized_keys` pada instans. Tempelkan informasi kunci publik dari pasangan kunci baru Anda di bawah informasi kunci publik yang sudah ada. Simpan file tersebut.
5. Putuskan sambungan dari instans Anda, dan uji apakah Anda dapat terhubung ke instans Anda menggunakan file kunci privat baru tersebut.
6. (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`.

 Important

Jika Anda menggunakan grup Auto Scaling, pastikan pasangan kunci yang Anda ganti tidak ditentukan dalam templat peluncuran atau konfigurasi peluncuran Anda. Jika Amazon EC2 Auto Scaling mendeteksi instans yang tidak sehat, ia akan meluncurkan instans pengganti. Namun demikian, peluncuran instans tersebut akan gagal jika pasangan kunci tidak dapat ditemukan. Untuk informasi selengkapnya, lihat [Templat peluncuran](#) di Panduan Pengguna Amazon EC2 Auto Scaling.

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

Setelah Anda menghapus semua kunci publik dari instans dan memutuskan sambungan dari instans tersebut, Anda tidak akan dapat terhubung lagi pada instans tersebut kecuali AMI menyediakan cara lain untuk login.

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 EC2 Linux Anda, bukan sidik jari key pair Anda, lihat [Dapatkan sidik jari instance](#).

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.

(Instance Linux) Fungsi hash yang 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 untuk menggunakan key pair yang sama untuk terhubung ke instance 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. Hal ini karena sidik jari dari kunci RSA asli yang dibuat menggunakan Amazon EC2 dikalkulasi menggunakan fungsi hash SHA-1, sedangkan sidik jari kunci RSA yang diimpor dikalkulasi menggunakan fungsi hash MD5.
- Untuk pasangan kunci ED25519, sidik jari akan tetap sama terlepas dari apakah pasangan kunci tersebut dibuat oleh Amazon EC2 atau diimpor ke Amazon EC2, karena fungsi hash SHA-256 yang sama digunakan untuk melakukan kalkulasi terhadap 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 yang ada di [Panduan Instalasi Windows 10](#). 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:

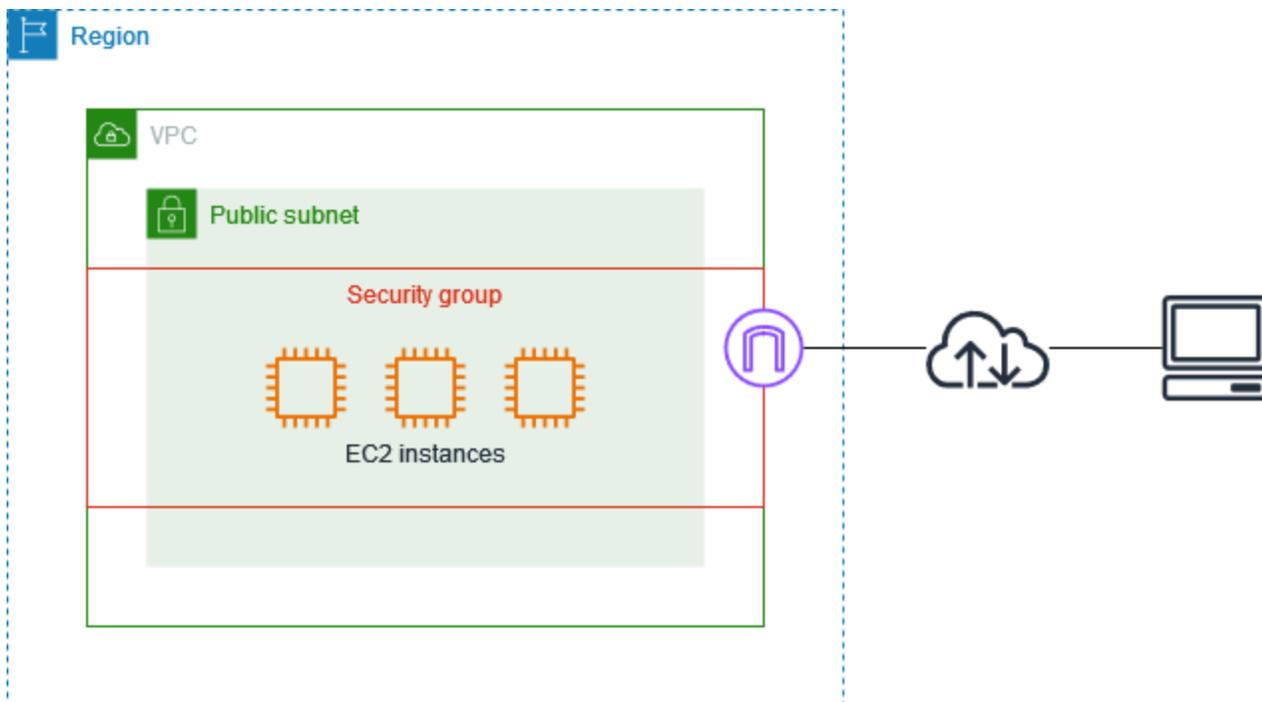
```
ssh-keygen -l -f path_to_private_key
```

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. Anda dapat menambahkan aturan-aturan ke setiap grup keamanan yang akan mengizinkan lalu lintas ke atau dari instans-instans yang dikaitkan. 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 untuk menjangkau instans, Amazon EC2 akan mengevaluasi semua aturan dari semua grup keamanan yang dikaitkan dengan instans tersebut.

Diagram berikut menunjukkan VPC dengan subnet, gateway internet, dan grup keamanan. Subnet berisi instans EC2. Grup keamanan ditugaskan ke instance. Satu-satunya lalu lintas yang mencapai instance adalah lalu lintas yang diizinkan oleh aturan grup keamanan. Misalnya, jika grup keamanan berisi aturan yang memungkinkan lalu lintas SSH dari jaringan Anda, maka Anda dapat terhubung ke instance Anda dari komputer menggunakan SSH. Jika grup keamanan berisi aturan yang memungkinkan semua lalu lintas dari sumber daya yang ditugaskan padanya, maka setiap instance dapat menerima lalu lintas apa pun yang dikirim dari instance lain.



Setelah Anda meluncurkan instans, Anda dapat mengubah grup keamanannya. Grup keamanan dikaitkan dengan antarmuka jaringan. Mengubah grup keamanan instans akan mengubah grup keamanan yang dikaitkan dengan antarmuka jaringan primer (eth0). Untuk informasi selengkapnya, lihat [Mengubah grup keamanan instans](#). Anda juga dapat mengubah grup keamanan yang

dikaitkan dengan antarmuka jaringan lainnya. Untuk informasi selengkapnya, lihat [Mengubah atribut antarmuka jaringan](#).

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Untuk informasi lebih lanjut, lihat [Keamanan dalam Amazon EC2](#). AWS menyediakan grup keamanan sebagai salah satu alat untuk mengamankan instans 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.

Tidak ada biaya tambahan untuk menggunakan grup keamanan.

Daftar Isi

- [Aturan-aturan grup keamanan](#)
- [Pelacakan koneksi grup keamanan](#)
- [Grup keamanan default dan kustom](#)
- [Cara menggunakan grup keamanan](#)
- [Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda](#)

Aturan-aturan grup keamanan

Aturan-aturan dari grup keamanan mengontrol lalu lintas ke dalam yang diperbolehkan untuk mencapai instans yang dikaitkan dengan grup keamanan. Aturan-aturan tersebut juga mengontrol lalu lintas ke luar yang diperbolehkan untuk meninggalkannya.

Berikut ini adalah karakteristik dari aturan-aturan grup keamanan:

- Secara default, grup keamanan berisi aturan keluar yang mengizinkan semua lalu lintas keluar. Anda dapat menghapus aturan ini. Perlu diperhatikan bahwa Amazon EC2 memblokir lalu lintas pada port 25 secara default. Untuk informasi selengkapnya, lihat [Pembatasan pada email yang dikirim menggunakan port 25](#).
- Aturan-aturan grup keamanan selalu bersifat permisif; Anda tidak dapat membuat aturan-aturan yang menolak akses.
- Aturan-aturan grup keamanan memungkinkan Anda untuk memfilter lalu lintas berdasarkan protokol dan jumlah port.
- 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. Untuk grup keamanan VPC, hal ini juga berarti tanggapan terhadap lalu lintas ke dalam yang diperbolehkan dapat mengalir ke luar, tanpa memedulikan aturan-aturan ke luar. Untuk informasi selengkapnya, lihat [Pelacakan koneksi grup keamanan](#).

- Anda dapat menambahkan dan menghapus aturan kapan saja. Perubahan-perubahan yang Anda buat akan diterapkan secara otomatis ke instans yang dikaitkan dengan grup keamanan.

Dampak dari beberapa perubahan aturan dapat bergantung pada cara pelacakan lalu lintas yang digunakan. Untuk informasi selengkapnya, lihat [Pelacakan koneksi grup keamanan](#).

- 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 akan menggunakan set aturan tersebut untuk menentukan apakah akses diperbolehkan atau tidak.

Anda dapat menetapkan beberapa grup keamanan pada instans. 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.

Note

[Grup keamanan tidak dapat memblokir permintaan DNS ke atau dari Resolver Route 53, kadang-kadang disebut sebagai 'alamat IP VPC+2' \(lihat \[Apa itu Amazon Route 53 Resolver?\]\(#\) di Panduan Pengembang Amazon Route 53\), atau 'AmazonProvidedDNS' \(lihat \[set opsi Bekerja dengan DHCP\]\(#\) 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).

Untuk setiap aturan, Anda harus menentukan hal-hal berikut:

- Name: Nama untuk grup keamanan (contohnya, “grup-keamanan-saya”).

nama dapat memiliki panjang hingga 255 karakter. Karakter yang diperbolehkan adalah a-z, A-Z, 0-9, spasi, dan `._-:/()#,@[]+=;{}!$*`. Saat nama tersebut berisi spasi di bagian paling belakang, kami akan menghapus spasi tersebut saat kami menyimpan nama tersebut. Sebagai contoh, jika Anda

memasukkan "Grup Keamanan Pengujian " sebagai namanya, maka kami menyimpannya sebagai "Grup Keamanan Pengujian".

- Protocol: Protokol yang akan diizinkan. Protokol yang paling umum adalah 6 (TCP), 17 (UDP), dan 1 (ICMP).
- Port range: Untuk TCP, UDP, atau protokol kustom, ada rentang port yang diizinkan. Anda dapat menentukan satu nomor port (misalnya, 22), atau rentang nomor port (misalnya, 7000-8000).
- Tipe dan kode ICMP: Untuk ICMP, jenis dan kode ICMP. Sebagai contoh, gunakan tipe 8 untuk ICMP Echo Request atau tipe 128 untuk ICMPv6 Echo Request.
- Source or destination: Sumber (aturan ke dalam) atau tujuan (aturan ke luar) untuk lalu lintas yang akan diizinkan. Tentukan satu dari yang berikut ini:
 - Satu alamat IPv4. Anda harus menggunakan panjang awalan /32. Sebagai contoh, 203.0.113.1/32.
 - Satu alamat IPv6. Anda harus menggunakan panjang awalan /128. Sebagai contoh, 2001:db8:1234:1a00::123/128.
 - Rentang alamat IPv4, dalam notasi blok CIDR. Sebagai contoh, 203.0.113.0/24.
 - Rentang alamat IPv6, dalam notasi blok CIDR. Sebagai contoh, 2001:db8:1234:1a00::/64.
 - ID daftar awalan. Sebagai contoh, p1-1234abc1234abc123. Untuk informasi lebih lanjut, lihat [Daftar awalan](#) di Panduan Pengguna Amazon VPC.
 - ID dari grup keamanan (di sini disebut sebagai grup keamanan yang ditentukan). Sebagai contoh, grup keamanan saat ini, grup keamanan dari VPC yang sama, atau grup keamanan untuk VPC yang di-peering. Hal ini memungkinkan lalu lintas berdasarkan alamat IP privat dari sumber daya yang dikaitkan dengan grup keamanan yang ditentukan. Hal ini tidak akan menambah aturan-aturan dari grup keamanan yang ditentukan ke grup keamanan saat ini.
- (Opsional) Deskripsi: Anda dapat menambahkan deskripsi untuk aturan, yang dapat membantu Anda untuk mengidentifikasinya nanti. deskripsi dapat memiliki panjang hingga 255 karakter. Karakter yang diperbolehkan adalah a-z, A-Z, 0-9, spasi, dan ._-:/()#,@[]+=;{}!\$*.

Saat Anda membuat aturan grup keamanan, AWS tetapkan ID unik ke aturan tersebut. Anda dapat menggunakan ID aturan tersebut ketika Anda menggunakan API atau CLI untuk mengubah atau menghapus aturan tersebut.

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](#). Jika aturan grup keamanan Anda mereferensikan grup keamanan yang sudah dihapus dalam VPC yang sama atau di VPC rekan, atau jika mereferensikan grup keamanan di VPC rekan yang koneksi peering VPC-nya telah dihapus, aturan tersebut akan ditandai sebagai kedaluwarsa. Untuk informasi selengkapnya, lihat [Cara Menggunakan Aturan Grup Keamanan Kedaluwarsa](#) dalam Panduan Peering VPC Amazon.

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 semua orang, maka setiap orang memiliki akses ke port TCP 22.

Saat Anda menambahkan, memperbarui atau menghapus aturan, perubahan-perubahan tersebut akan diterapkan secara otomatis pada semua instans yang dikaitkan dengan grup keamanan.

Pelacakan koneksi grup keamanan

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 \[set opsi Bekerja dengan DHCP\]\(#\) 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 arus 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. Sebagai contoh, jika Anda memiliki aturan ke luar (0.0.0.0/0) yang terbuka dan Anda menghapus aturan yang mengizinkan semua lalu lintas SSH ke dalam (0.0.0.0/0) (port TCP 22) ke instans (atau memodifikasinya sehingga koneksi tidak diizinkan lagi), maka koneksi SSH yang sudah ada pada instans tersebut akan langsung dibuang. Koneksi tersebut sebelumnya tidak dilacak, sehingga perubahan yang diterapkan akan memutuskan 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 dilakukan melalui berikut ini secara otomatis dilacak, bahkan jika konfigurasi grup keamanan tidak memerlukan pelacakan:

- Gateway internet khusus egress
- 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)

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 performa jaringan untuk 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.

Pertimbangan kinerja pelacakan koneksi

Perutean asimetris, di mana lalu lintas masuk ke sebuah instance melalui satu antarmuka jaringan dan pergi melalui antarmuka jaringan yang berbeda, dapat mengurangi kinerja puncak yang dapat dicapai oleh instans jika arus dilacak.

Untuk mempertahankan kinerja puncak saat pelacakan koneksi diaktifkan untuk grup keamanan Anda, kami merekomendasikan konfigurasi berikut:

- Hindari topologi routing asimetris, jika memungkinkan.
- Alih-alih menggunakan grup keamanan untuk memfilter, gunakan ACL jaringan.
- Jika Anda harus menggunakan grup keamanan dengan pelacakan koneksi, konfigurasi batas waktu koneksi sesingkat mungkin.

Untuk informasi selengkapnya tentang penyetelan kinerja pada sistem Nitro, lihat [Pertimbangan 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 terbebannya 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 untuk [instance yang dibangun di atas Sistem AWS Nitro](#).

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 (5 hari). Default: 432.000 detik. 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 (3 menit). 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 `contrack_allowance_exceeded` dan `contrack_allowance_available` memungkinkan Anda memantau paket yang terputus dan pemanfaatan koneksi yang dilacak untuk secara proaktif mengelola kapasitas instans EC2 dengan tindakan peningkatan atau penurunan skala untuk membantu memenuhi permintaan koneksi jaringan sebelum memutuskan paket. Jika Anda mengamati pemutusan `contrack_allowance_exceeded` pada instans EC2, Anda dapat memperoleh manfaat dari mengatur batas waktu yang ditetapkan TCP yang lebih rendah untuk menghitung sesi TCP/UDP lama yang dihasilkan dari klien yang tidak tepat atau kotak tengah jaringan.
- 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 perutean asimetris, kami sarankan Anda mengonfigurasi batas waktu idle yang ditetapkan TCP selama 60 detik.
- Jika Anda menjalankan beban kerja dengan jumlah koneksi yang banyak seperti DNS, SIP, SNMP, Syslog, Radius, dan layanan lain yang utamanya menggunakan UDP untuk melayani permintaan, mengatur batas waktu 'aliran UDP' ke 60 detik akan memberikan skala/performa yang lebih tinggi untuk kapasitas yang ada serta untuk mencegah kegagalan gray.
- Untuk koneksi TCP/UDP melalui penyeimbang beban jaringan (NLB) dan penyeimbang beban elastis (ELB), semua koneksi akan 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 guna memungkinkan lebih banyak fleksibilitas untuk batas waktu daripada default untuk ELB/NLB.

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 ke dalam dan ke luar pada port 22 (SSH) akan dilacak, karena aturan ke dalam hanya mengizinkan lalu lintas dari 203.0.113.1/32 saja, dan bukan 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.

Grup keamanan default dan kustom

AWS Akun Anda secara otomatis memiliki grup keamanan default untuk VPC default di setiap Wilayah. Jika Anda tidak menentukan grup keamanan saat meluncurkan instans, maka instans Anda secara otomatis akan dikaitkan dengan grup keamanan default untuk VPC. Jika Anda tidak ingin instans Anda menggunakan grup keamanan default, maka Anda dapat membuat grup keamanan kustom Anda sendiri dan menentukannya saat peluncuran instans Anda.

Daftar Isi

- [Grup keamanan default](#)
- [Grup keamanan kustom](#)

Grup keamanan default

Setiap VPC dilengkapi dengan grup keamanan default. Kami merekomendasikan agar Anda membuat grup keamanan untuk instans atau grup instans tertentu, alih-alih menggunakan grup keamanan default. Namun, jika Anda tidak menentukan grup keamanan saat meluncurkan instans, maka kami mengaitkan instans tersebut dengan grup keamanan default untuk VPC.

Nama grup keamanan default adalah "default". Berikut ini adalah aturan default untuk grup keamanan default.

Jalur masuk

Sumber	Protokol	Rentang Port	Deskripsi
<i>sg-1234567890abcde</i> <i>f0</i>	Semua	Semua	Mengizinkan lalu lintas jalur masuk dari semua sumber daya yang ditetapkan untuk grup keamanan ini. Sumbernya adalah ID dari grup keamanan ini.

Jalur keluar

Tujuan	Protokol	Rentang Port	Deskripsi
0.0.0.0/0	Semua	Semua	Mengizinkan semua lalu lintas IPv4 ke luar.
::/0	Semua	Semua	Mengizinkan semua lalu lintas IPv6 ke luar. Aturan ini ditambahkan hanya jika VPC Anda memiliki blok CIDR IPv6 yang dikaitkan.

Dasar-dasar grup keamanan default

- Anda dapat mengubah aturan untuk grup keamanan default.
- Anda tidak dapat menghapus grup keamanan default. Jika Anda mencoba menghapus grup keamanan default, kami akan menampilkan kode kesalahan berikut: `Client.CannotDelete`.

Grup keamanan kustom

Anda dapat membuat banyak grup keamanan untuk merefleksikan berbagai peran yang dijalankan oleh instans Anda; misalnya, server web atau server basis data.

Saat Anda membuat grup keamanan, Anda harus memberi nama dan deskripsi untuknya. Nama dan deskripsi grup keamanan dapat memiliki panjang hingga 255 karakter, dan terbatas hanya pada karakter-karakter berikut ini:

a-z, A-Z, 0-9, spasi, dan `._-:/()#,@[]+=&:{}!$*`

Nama grup keamanan tidak dapat dimulai dengan yang berikut: `sg-`. Nama grup keamanan harus unik untuk VPC.

Berikut ini adalah aturan-aturan default untuk grup keamanan yang Anda buat:

- Tidak mengizinkan lalu lintas ke dalam
- Mengizinkan semua lalu lintas ke luar

Setelah Anda membuat grup keamanan, Anda dapat mengubah aturan-aturan ke dalam grup keamanan tersebut untuk mencerminkan jenis lalu lintas ke dalam yang Anda inginkan untuk menjangkau instans-instans yang dikaitkan. Anda juga dapat mengubah aturan-aturan ke luarnya.

Untuk informasi selengkapnya tentang aturan-aturan yang dapat Anda tambahkan ke grup keamanan, lihat [Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda](#).

Cara menggunakan grup keamanan

Anda dapat menetapkan grup keamanan ke instans saat Anda meluncurkan instans tersebut. Saat Anda menambahkan atau menghapus aturan, perubahan tersebut akan diterapkan secara otomatis ke semua instans yang memiliki grup keamanan yang telah Anda tetapkan. Untuk informasi selengkapnya, lihat [Menetapkan grup keamanan pada instans](#).

Setelah Anda meluncurkan instans, Anda dapat mengubah grup keamanannya. Untuk informasi selengkapnya, lihat [Mengubah grup keamanan instans](#).

Anda dapat membuat, melihat, memperbarui, dan menghapus grup keamanan dan aturan-aturan dari grup keamanan tersebut menggunakan konsol Amazon EC2 dan alat baris perintah.

Tugas

- [Membuat grup keamanan](#)
- [Menyalin grup keamanan](#)
- [Menampilkan grup keamanan Anda](#)
- [Menambahkan aturan ke grup keamanan](#)
- [Memperbarui aturan-aturan grup keamanan](#)
- [Menghapus aturan dari grup keamanan](#)
- [Menghapus grup keamanan](#)
- [Menetapkan grup keamanan pada instans](#)
- [Mengubah grup keamanan instans](#)

Membuat grup keamanan

Meskipun Anda dapat menggunakan grup keamanan default untuk instans Anda, Anda mungkin ingin membuat grup keamanan Anda sendiri untuk mencerminkan berbagai peran berbeda yang dimainkan oleh instans-instans tersebut dalam sistem Anda.

Secara default, grup keamanan baru dimulai hanya dengan aturan ke luar yang mengizinkan semua lalu lintas meninggalkan instans. Anda harus menambahkan aturan-aturan lain untuk mengizinkan lalu lintas ke dalam atau membatasi lalu lintas ke luar.

grup keamanan hanya dapat digunakan dalam VPC yang menjadi tujuan pembuatan grup keamanan tersebut.

Console

Cara membuat 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 Create security group (Buat grup keamanan).
4. Dalam bagian Detail dasar, lakukan hal berikut.
 - a. Masukkan nama deskriptif dan deskripsi singkat untuk grup keamanan. Nama dan deskripsi tersebut tidak dapat diedit setelah grup keamanan dibuat. Nama dan deskripsi dapat memiliki panjang hingga 255 karakter. Karakter yang bisa digunakan adalah a-z, A-Z, 0-9, spasi, dan `._-:/()#,@[]+=&:{}!$*`.
 - b. Untuk VPC, pilih VPC.
5. Anda dapat menambahkan aturan-aturan grup keamanan sekarang, atau Anda dapat menambahkannya nanti. Untuk informasi selengkapnya, lihat [Menambahkan aturan ke grup keamanan](#).
6. Anda dapat menambahkan tanda sekarang, atau Anda dapat menambahkannya nanti. Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
7. Pilih Create security group (Buat grup keamanan).

Command line

Cara membuat grup keamanan

Gunakan salah satu perintah berikut:

- [create-security-group](#) (AWS CLI)
- [New-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

Menyalin grup keamanan

Anda dapat membuat grup keamanan baru dengan membuat salinan dari grup keamanan yang sudah ada. Saat Anda menyalin grup keamanan, salinan tersebut dibuat dengan aturan ke dalam dan ke luar yang sama seperti grup keamanan yang asli. Jika grup keamanan asli berada di VPC, maka salinannya akan dibuat dalam VPC yang sama kecuali Anda menentukan lain.

Salinan tersebut akan menerima ID grup keamanan unik dan Anda harus memberinya nama. Anda juga dapat menambahkan deskripsi.

Anda tidak dapat menyalin grup keamanan dari satu Wilayah ke Wilayah lain.

Anda dapat membuat salinan grup keamanan menggunakan konsol Amazon EC2.

Cara menyalin 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 yang akan disalin lalu pilih Actions (Tindakan), Copy to new security group (Salin ke grup keamanan baru).
4. Tentukan nama dan deskripsi opsional, dan ubah VPC dan aturan-aturan grup keamanan tersebut jika diperlukan.
5. Pilih Create (Buat).

Menampilkan grup keamanan Anda

Anda dapat melihat informasi tentang grup keamanan Anda menggunakan salah satu metode berikut.

Console

Cara menampilkan grup keamanan Anda

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pada panel navigasi, pilih Security Groups (Grup Keamanan).
3. Grup keamanan Anda sudah tercantum. Untuk melihat detail grup keamanan tertentu, termasuk aturan-aturan ke dalam dan ke luar yang dimilikinya, pilih ID dalam kolom ID Grup Keamanan.

Command line

Cara menampilkan grup keamanan Anda

Gunakan salah satu perintah berikut ini.

- [describe-security-groups](#) (AWS CLI)
- [describe-security-group-rules](#) (AWS CLI)
- [Get-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

Amazon EC2 Global View

Anda dapat menggunakan Amazon EC2 Global View untuk melihat grup keamanan Anda di semua Wilayah yang mengaktifkan AWS akun Anda. Untuk informasi selengkapnya, lihat [Amazon EC2 Global View](#).

Menambahkan aturan ke grup keamanan

Saat Anda menambahkan aturan ke grup keamanan, aturan yang baru itu akan secara otomatis diterapkan ke setiap instans yang dikaitkan dengan grup keamanan tersebut. Mungkin ada penundaan singkat sebelum aturan diterapkan. Untuk informasi selengkapnya, lihat [Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda](#) dan [Aturan-aturan grup keamanan](#).

Console

Cara menambahkan aturan ke dalam ke 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, dan kemudian pilih Actions (Tindakan), Edit inbound rules (Edit aturan ke dalam).
4. Untuk setiap aturan, pilih Add rule (Tambahkan aturan) dan lakukan hal-hal berikut.
 - a. Untuk Tipe, pilih jenis protokol yang diizinkan.
 - Untuk TCP Kustom atau UDP Kustom, Anda harus memasukkan rentang port untuk mengizinkan. Misalnya, 0-99.

- Untuk Custom ICMP, Anda harus memilih jenis ICMP dari Protocol. Rentang port dikonfigurasi untuk Anda. Sebagai contoh, untuk mengizinkan perintah ping, pilih Echo Request (Permintaan Echo) dari Protokol.
 - Untuk jenis lainnya, protokol dan rentang port akan dikonfigurasi untuk Anda.
- b. Untuk Sumber, lakukan salah satu hal berikut untuk mengizinkan lalu lintas.
- Pilih Kustom dan kemudian masukkan alamat IP dalam notasi CIDR, blok CIDR, grup keamanan lainnya, atau daftar awalan.
 - Pilih Anywhere untuk mengizinkan semua lalu lintas untuk protokol yang ditentukan yang akan mencapai instans Anda. Opsi ini secara otomatis akan menambahkan blok CIDR 0.0.0.0/0 IPv4 sebagai sumber. Jika grup keamanan Anda berada di VPC yang diizinkan untuk IPv6, maka opsi ini secara otomatis akan menambahkan aturan untuk blok CIDR ::/0 IPv6.
-  **Warning**

Jika memilih Di mana saja, Anda akan memungkinkan semua alamat IPv4 dan IPv6 mengakses instans dengan protokol tertentu. Jika menambahkan aturan untuk port 22 (SSH) atau 3389 (RDP), Anda harus mengotorisasi alamat IP tertentu saja atau rentang alamat untuk mengakses instans Anda.
- Pilih My IP (IP Saya) untuk mengizinkan lalu lintas ke dalam hanya dari alamat IPv4 publik komputer lokal Anda.
- c. Untuk Deskripsi, secara opsional Anda bisa menentukan deskripsi singkat untuk aturan.
5. Pilih Preview changes (Tinjau perubahan), Save rules (Simpan aturan).

Cara menambahkan aturan ke luar ke 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, dan kemudian pilih Actions (Tindakan), Edit outbound rules (Edit aturan ke luar).
4. Untuk setiap aturan, pilih Add rule (Tambahkan aturan) dan lakukan hal-hal berikut.
 - a. Untuk Tipe, pilih jenis protokol yang diizinkan.

- Untuk TCP Kustom atau UDP Kustom, Anda harus memasukkan rentang port untuk mengizinkan. Misalnya, 0-99.
 - Untuk Custom ICMP, Anda harus memilih jenis ICMP dari Protocol. Rentang port dikonfigurasi untuk Anda.
 - Untuk jenis lainnya, protokol dan rentang port akan dikonfigurasi secara otomatis.
- b. Untuk Tujuan, lakukan salah satu hal berikut ini.

- Pilih Kustom dan kemudian masukkan alamat IP dalam notasi CIDR, blok CIDR, grup keamanan lainnya, atau daftar awalan untuk mengizinkan lalu lintas ke luar.
- Pilih Anywhere untuk mengizinkan lalu lintas ke luar ke semua alamat IP. Opsi ini secara otomatis akan menambahkan blok CIDR 0.0.0.0/0 IPv4 sebagai tujuan.

Jika grup keamanan Anda berada di VPC yang diizinkan untuk IPv6, maka opsi ini secara otomatis akan menambahkan aturan untuk blok CIDR ::/0 IPv6.

- Pilih My IP (IP Saya) untuk mengizinkan lalu lintas ke luar hanya ke alamat IPv4 publik komputer lokal Anda saja.

- c. (Opsional) Untuk Deskripsi, tentukan deskripsi singkat untuk aturan.

5. Pilih Preview changes (Tinjau perubahan), Confirm (Mengonfirmasi).

Command line

Cara menambahkan aturan ke grup keamanan

Gunakan salah satu perintah berikut ini.

- [authorize-security-group-ingress](#) (AWS CLI)
- [Grant-EC2SecurityGroupIngress](#) (AWS Tools for Windows PowerShell)

Untuk menambahkan satu atau beberapa aturan egress ke grup keamanan

Gunakan salah satu perintah berikut ini.

- [authorize-security-group-egress](#) (AWS CLI)
- [Grant-EC2SecurityGroupEgress](#) (AWS Tools for Windows PowerShell)

Memperbarui aturan-aturan grup keamanan

Anda dapat memperbarui aturan grup keamanan menggunakan salah satu metode berikut. Aturan yang diperbarui secara otomatis akan diterapkan ke instans yang dikaitkan dengan grup keamanan.

Console

Saat Anda melakukan modifikasi pada protokol, rentang port, atau sumber atau tujuan dari aturan grup keamanan yang sudah ada menggunakan konsol, konsol tersebut akan menghapus aturan yang sudah ada dan menambahkan yang baru untuk Anda.

Cara memperbarui 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. Pilih Tindakan, Edit aturan ke dalam untuk memperbarui aturan untuk lalu lintas masuk atau Tindakan, Edit aturan ke luar untuk memperbarui aturan untuk lalu lintas ke luar.
5. Perbarui aturan sesuai yang dibutuhkan.
6. Pilih Preview changes (Tinjau perubahan), Confirm (Mengonfirmasi).

Cara memberikan tanda ke 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. Pada tab Inbound rules (Aturan ke dalam) atau Outbound rules (Aturan ke luar), pilih kotak centang untuk aturan dan kemudian pilih Manage tandas (Kelola tanda).
5. Halaman Kelola tanda akan menampilkan tanda yang ditetapkan ke aturan tersebut. Untuk menambahkan tanda, pilih Add tanda (Tambahkan tanda) dan masukkan kunci dan nilai tanda. Untuk menghapus tanda, pilih Remove (Hapus) yang ada di samping tanda yang ingin Anda hapus.
6. Pilih Save changes (Simpan perubahan).

Command line

Anda tidak dapat melakukan modifikasi terhadap protokol, rentang port, atau sumber atau tujuan dari aturan yang sudah ada menggunakan Amazon EC2 API atau alat baris perintah. Sebaliknya, Anda harus menghapus aturan yang sudah ada dan menambahkan aturan yang baru. Namun demikian, Anda dapat memperbarui deskripsi dari aturan yang sudah ada.

Cara memperbarui aturan

Gunakan salah satu perintah berikut ini.

- [modify-security-group-rules](#) (AWS CLI)

Untuk memperbarui deskripsi dari aturan ke dalam yang sudah ada

Gunakan salah satu perintah berikut ini.

- [update-security-group-rule-descriptions-ingress](#) (AWS CLI)
- [Update-EC2SecurityGroupRuleIngressDescription](#) (AWS Tools for Windows PowerShell)

Untuk memperbarui deskripsi dari aturan ke luar yang sudah ada

Gunakan salah satu perintah berikut ini.

- [update-security-group-rule-descriptions-egress](#) (AWS CLI)
- [Update-EC2SecurityGroupRuleEgressDescription](#) (AWS Tools for Windows PowerShell)

Cara memberikan tanda ke aturan grup keamanan

Gunakan salah satu perintah berikut ini.

- [create-tags](#) (AWS CLI)
- [New-EC2Tag](#) (AWS Tools for Windows PowerShell)

Menghapus aturan dari grup keamanan

Saat Anda menghapus aturan dari grup keamanan, perubahan tersebut secara otomatis akan diterapkan pada setiap instans yang dikaitkan dengan grup keamanan.

Anda dapat menghapus aturan-aturan dari grup keamanan menggunakan salah satu metode berikut.

Console

Cara menghapus 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 yang akan diperbarui, pilih Actions (Tindakan), dan kemudian pilih Edit inbound rules (Edit aturan ke dalam) untuk menghapus satu aturan ke dalam atau Edit outbound rules (Edit aturan ke luar) untuk menghapus satu aturan ke luar.
4. Pilih tombol Delete (Hapus) yang ada di sebelah kanan aturan yang akan dihapus.
5. Pilih Simpan aturan. Atau, pilih Pratinjau perubahan, tinjau perubahan Anda, dan pilih Konfirmasi.

Command line

Untuk menghapus satu atau beberapa aturan ingress dari grup keamanan

Gunakan salah satu perintah berikut ini.

- [revoke-security-group-ingress](#) (AWS CLI)
- [Revoke-EC2SecurityGroupIngress](#) (AWS Tools for Windows PowerShell)

Untuk menghapus satu atau beberapa aturan egress dalam dari grup keamanan

Gunakan salah satu perintah berikut ini.

- [revoke-security-group-egress](#) (AWS CLI)
- [Revoke-EC2SecurityGroupEgress](#) (AWS Tools for Windows PowerShell)

Menghapus grup keamanan

Anda tidak dapat menghapus grup keamanan yang dikaitkan dengan instans. Anda tidak dapat menghapus grup keamanan default. Anda tidak dapat menghapus grup keamanan yang dirujuk oleh aturan dalam grup keamanan lain dalam VPC yang sama. Jika grup keamanan Anda dirujuk oleh salah satu aturannya sendiri, Anda harus menghapus aturan itu sebelum Anda dapat menghapus grup keamanan tersebut.

Console

Cara menghapus grup keamanan

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Pada panel navigasi, pilih Grup Keamanan.
3. Pilih grup keamanan lalu pilih Tindakan, Hapus Grup Keamanan.
4. Saat diminta konfirmasi, pilih Hapus.

Command line

Cara menghapus grup keamanan

Gunakan salah satu perintah berikut ini.

- [delete-security-group](#) (AWS CLI)
- [Remove-EC2SecurityGroup](#) (AWS Tools for Windows PowerShell)

Menetapkan grup keamanan pada instans

Anda dapat menetapkan satu atau beberapa grup keamanan ke instans saat Anda meluncurkan instans tersebut. Anda juga dapat menentukan satu atau beberapa grup keamanan di templat peluncuran. Grup keamanan akan ditetapkan ke semua instans yang diluncurkan menggunakan templat peluncuran.

- Untuk menetapkan grup keamanan ke instans saat Anda meluncurkan instans, lihat [Pengaturan jaringan](#) dari [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#) (konsol baru) atau [Langkah 6: Konfigurasi Grup Keamanan](#) (konsol lama).
- Untuk menentukan grup keamanan di templat peluncuran, lihat [Pengaturan jaringan](#) dari [Buat template peluncuran dari parameter](#).

Mengubah grup keamanan instans

Setelah Anda meluncurkan instans, Anda dapat mengubah grup keamanannya dengan menambahkan atau menghapus grup keamanan tersebut.

Persyaratan

- Instans harus berada dalam status `running` atau `stopped`.
- Grup keamanan bersifat khusus untuk VPC. Anda dapat menetapkan grup keamanan ke satu atau beberapa instans yang diluncurkan di VPC tempat Anda membuat grup keamanan.

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).

Command line

Untuk mengubah grup keamanan instans

Gunakan salah satu perintah berikut ini.

- [modify-instance-attribute](#) (AWS CLI)
- [Edit-EC2InstanceAttribute](#) (AWS Tools for Windows PowerShell)

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

mempunyai 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-aturan untuk melakukan ping/ICMP](#)
- [Aturan-aturan server DNS](#)
- [Aturan-aturan Amazon EFS](#)
- [Aturan-aturan Penyeimbangan Beban Elastis](#)
- [Aturan-aturan peering VPC](#)

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	(Khusus VPC yang diaktifkan IPv6) Mengizinkan akses HTTP ke luar ke alamat IPv6 mana pun
TCP	6	443 (HTTPS)	::/0	(Khusus VPC yang diaktifkan IPv6) Mengizinkan akses HTTPS ke luar ke alamat IPv6 mana 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-aturan untuk melakukan 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		

Tipe	Protokol	Sumber		
		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 menggunakan penyeimbang beban, maka grup keamanan yang dikaitkan dengan penyeimbang beban Anda harus memiliki aturan-aturan yang mengizinkan komunikasi dengan instans atau target Anda. Untuk informasi selengkapnya, lihat [mengonfigurasi grup keamanan untuk Penyeimbang Beban Klasik Anda](#) dalam Panduan Pengguna untuk Penyeimbang Beban Klasik,

dan [Grup Keamanan untuk Penyeimbang Beban Aplikasi Anda](#) dalam Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Aturan-aturan peering VPC

Anda dapat memperbarui aturan-aturan ke dalam atau ke luar untuk grup keamanan VPC Anda untuk mereferensikan grup keamanan di VPC yang tersambung. Dengan melakukan hal itu, lalu lintas dapat mengalir ke dan dari instans yang dikaitkan dengan grup keamanan yang dirujuk di VPC yang tersambung. Untuk informasi selengkapnya tentang cara mengonfigurasi grup keamanan untuk peering VPC, lihat [Memperbarui grup keamanan Anda untuk mereferensikan grup VPC rekan](#).

NitroTPM

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 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 [instance yang dibangun](#) pada Sistem Nitro. AWS Anda dapat memilih salah satu AMI yang telah dibuat sebelumnya oleh Amazon atau Anda dapat membuatnya sendiri.

Biaya

Tidak ada biaya tambahan yang dikenakan untuk menggunakan NitroTPM. Anda hanya harus membayar untuk sumber daya dasar yang Anda gunakan.

Topik

- [Pertimbangan-pertimbangan](#)
- [Prasyarat untuk mengaktifkan saat peluncuran](#)
- [Membuat AMI untuk dukungan NitroTPM](#)
- [Verifikasi apakah AMI sudah diaktifkan untuk NitroTPM](#)
- [Mengaktifkan atau menghentikan menggunakan NitroTPM pada instans](#)
- [Ambil kunci dukungan publik untuk sebuah contoh](#)

Pertimbangan-pertimbangan

Pertimbangan-pertimbangan berikut ini berlaku saat Anda menggunakan NitroTPM:

- BitLocker volume yang dienkrpsi dengan kunci berbasis NitroTPM hanya dapat digunakan pada instance asli.
- Status NitroTPM tidak disertakan dalam [Snapshot Amazon EBS](#).
- Status NitroTPM tidak disertakan dalam citra [VM Import/Export](#).
- Dukungan NitroTPM diaktifkan dengan menentukan nilai `v2.0` untuk parameter `tpm-support` saat Anda membuat AMI. Setelah Anda meluncurkan instans dengan AMI tersebut, Anda tidak dapat melakukan modifikasi terhadap atribut yang ada pada instans. [Instance dengan nitRotPM tidak mendukung API Atribut. ModifyInstance](#)
- Anda hanya dapat membuat AMI dengan NITroTPM yang dikonfigurasi dengan menggunakan [RegisterImage](#)API dengan menggunakan AWS CLI dan bukan dengan konsol Amazon EC2.
- NitroTPM tidak didukung pada Outposts.
- NitroTPM tidak didukung di Zona Lokal atau Zona Wavelength.

Prasyarat untuk mengaktifkan saat peluncuran

Untuk meluncurkan instance dengan NitroTPM diaktifkan, prasyarat berikut harus ada.

Instans Linux

AMI

Mengharuskan AMI dengan NitroTPM yang diaktifkan.

Saat ini, tidak ada AMI Amazon Linux yang diaktifkan NitroTPM. Agar dapat menggunakan AMI yang didukung, Anda harus melakukan sejumlah langkah konfigurasi pada Linux AMI Anda sendiri. Untuk informasi selengkapnya, lihat [Membuat AMI untuk dukungan NitroTPM](#).

Sistem operasi

AMI harus menyertakan sistem operasi dengan driver TPM 2.0 CRB. Sebagian besar sistem operasi saat ini, seperti Amazon Linux 2, berisi driver TPM 2.0 CRB.

Mode booting UEFI

NitroTPM mengharuskan instans berjalan dalam mode booting UEFI, yang mewajibkan AMI harus dikonfigurasi untuk mode booting UEFI. Untuk informasi selengkapnya, lihat [UEFI Secure Boot](#).

Instans Windows

AMI

Mengharuskan AMI dengan NitroTPM yang diaktifkan.

AMI Windows berikut telah dikonfigurasi sebelumnya untuk mengaktifkan NitroTPM dan UEFI Secure Boot dengan kunci Microsoft:

- TPM-Windows_Server-2022-English-Core-Base
- TPM-Windows_Server-2022-English-Full-Base
- TPM-Windows_Server-2022-English-Full-SQL_2022_Enterprise
- TPM-Windows_Server-2022-English-Full-SQL_2022_Standard
- TPM-Windows_Server-2019-English-Core-Base
- TPM-Windows_Server-2019-English-Full-Base
- TPM-Windows_Server-2019-Inggris-penuh-SQL_2019_Enterprise
- TPM-Windows_Server-2019-English-Full-SQL_2019_Standard
- TPM-Windows_Server-2016-English-Core-Base
- TPM-Windows_Server-2016-English-Full-Base

Saat ini, kami tidak mendukung impor Windows dengan NitroTPM menggunakan perintah [import-image](#).

Sistem operasi

AMI harus menyertakan sistem operasi dengan driver TPM 2.0 CRB. Sebagian besar sistem operasi saat ini, seperti TPM-Windows_Server-2022-English-full-base, berisi driver TPM 2.0 CRB.

Mode booting UEFI

NitroTPM mengharuskan instans berjalan dalam mode booting UEFI, yang mewajibkan AMI harus dikonfigurasi untuk mode booting UEFI. Untuk informasi selengkapnya, lihat [UEFI Secure Boot](#).

Tipe instans

Anda harus menggunakan salah satu jenis instance virtual berikut:

- Tujuan umum: M5, M5a, M5ad, M5d, M5dn, M5n, M5zn, M6a, M6i, M6iD, M6idn, M6in, M7a, M7i, M7i-flex, T3, T3a
- Komputasi dioptimalkan: C5, C5a, C5ad, C5d, C5n, C6a, C6i, C6id, C6in, C7a, C7i, C7i-flex
- Memori dioptimalkan: R5, R5a, R5ad, R5b, R5d, R5dn, R5n, R6a, R6i, R6iDn, R6in, R6iD, R7a, R7i, R7iZ, U7i-12TB, U7in-16TB, U7in-24TB, U7in-32TB, x2idn, X2idn, 2iEDN, X2IEZN, z1d
- Penyimpanan dioptimalkan: D3, D3en, i3en, i4i
- Komputasi yang dipercepat: G4dn, G5, G6, Gr6, Inf1, Inf2
- Komputasi kinerja tinggi: HPC6a, HPC6id

Note

Instans berbasis graviton, instance Xen, instance Mac, dan instance bare metal tidak didukung.

Membuat AMI untuk dukungan NitroTPM

Anda harus mengonfigurasi AMI Linux untuk dukungan NitroTPM saat mendaftarkan AMI. Anda tidak dapat mengonfigurasi dukungan NitroTPM setelahnya.

Untuk daftar AMI Windows yang telah dikonfigurasi sebelumnya untuk dukungan NitroTPM, lihat [Prasyarat untuk mengaktifkan saat peluncuran](#)

Untuk mendaftarkan AMI Linux untuk dukungan NitroTPM

1. Luncurkan instance sementara dengan AMI Linux yang Anda butuhkan.
2. Setelah instance mencapai `running` status, buat snapshot dari volume root instance.
3. Daftarkan AMI baru. Gunakan perintah [register-image](#). Untuk `--tpm-support`, tentukan `v2.0`. Untuk `--boot-mode`, tentukan `uefi`. Dan tentukan pemetaan perangkat blok untuk volume root menggunakan snapshot yang Anda buat pada langkah sebelumnya.

```
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=snapshot_id} \  
  --tpm-support v2.0
```

Output yang diharapkan

```
{  
  "ImageId": "ami-0123456789example"  
}
```

4. Hentikan instance sementara yang Anda luncurkan di langkah 1, jika tidak lagi diperlukan.

Verifikasi apakah AMI sudah diaktifkan untuk NitroTPM

Anda dapat menggunakan `describe-images` atau `describe-image-attributes` untuk memverifikasi apakah AMI sudah diaktifkan untuk NitroTPM.

Cara melakukan verifikasi apakah AMI sudah diaktifkan untuk NitroTPM menggunakan **describe-images**

Gunakan perintah [describe-images](#) dan tentukan ID AMI.

```
aws ec2 describe-images --image-ids ami-0123456789example
```

Jika NitroTPM sudah diaktifkan untuk AMI, `"TpmSupport": "v2.0"` akan muncul di output.

```
{  
  "Images": [  
    {  
      "ImageId": "ami-0123456789example",  
      "TpmSupport": "v2.0"  
    }  
  ]  
}
```

```
{
  ...
  "BootMode": "uefi",
  ...
  "TpmSupport": "v2.0"
}
]
```

Cara melakukan verifikasi apakah AMI sudah diaktifkan untuk NitroTPM menggunakan **describe-image-attribute**

Gunakan perintah [describe-image-attribute](#) dan tentukan parameter `attribute` dengan nilai `tpmSupport`.

Note

Anda harus menjadi pemilik AMI untuk memanggil `describe-image-attribute`.

```
aws ec2 describe-image-attribute \
  --region us-east-1 \
  --image-id ami-0123456789example \
  --attribute tpmSupport
```

Jika NitroTPM sudah diaktifkan untuk AMI, nilai untuk `TpmSupport` adalah `"v2.0"`. Perhatikan bahwa `describe-image-attribute` hanya mengembalikan atribut-atribut yang ditentukan dalam permintaan.

```
{
  "ImageId": "ami-0123456789example",
  "TpmSupport": {
    "Value": "v2.0"
  }
}
```

Mengaktifkan atau menghentikan menggunakan NitroTPM pada instans

Saat Anda meluncurkan instans dari AMI yang memiliki dukungan NitroTPM yang sudah diaktifkan, instans tersebut akan diluncurkan dengan NitroTPM yang diaktifkan. Anda dapat mengonfigurasi

instans untuk berhenti menggunakan NitroTPM. Anda dapat melakukan verifikasi apakah instans sudah diaktifkan untuk NitroTPM atau tidak.

Topik

- [Meluncurkan instans dengan NitroTPM](#)
- [Menghentikan menggunakan NitroTPM pada instans](#)
- [Lakukan verifikasi apakah NitroTPM dapat diakses di dalam instans](#)

Meluncurkan instans dengan NitroTPM

Ketika Anda meluncurkan instans dengan [prasyarat](#), NitroTPM secara otomatis akan diaktifkan pada instans. Anda hanya dapat mengaktifkan NitroTPM pada instans di saat peluncuran. Untuk informasi tentang cara meluncurkan instans, lihat [Luncurkan instans Anda](#).

Menghentikan menggunakan NitroTPM pada instans

Setelah meluncurkan instans dengan NitroTPM yang diaktifkan, Anda tidak dapat menonaktifkan NitroTPM untuk instans 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:

- [Instans Linux] Gunakan tpm-tools.
- [Instans Windows] Gunakan konsol manajemen TPM, tpm.msc.

Untuk informasi selengkapnya tentang bagaimana menonaktifkan driver perangkat, lihat dokumentasi untuk sistem operasi Anda.

Lakukan verifikasi apakah NitroTPM dapat diakses di dalam instans

Untuk memverifikasi apakah sebuah instance diaktifkan untuk dukungan NitroTPM menggunakan AWS CLI

Gunakan perintah [describe-instances](#) AWS CLI dan tentukan ID instans. Saat ini, konsol Amazon EC2 tidak menampilkan bidang TpmSupport.

```
aws ec2 describe-instances --instance-ids i-0123456789example
```

Jika dukungan NitroTPM sudah diaktifkan pada instans tersebut, "TpmSupport": "v2.0" akan muncul di output-nya.

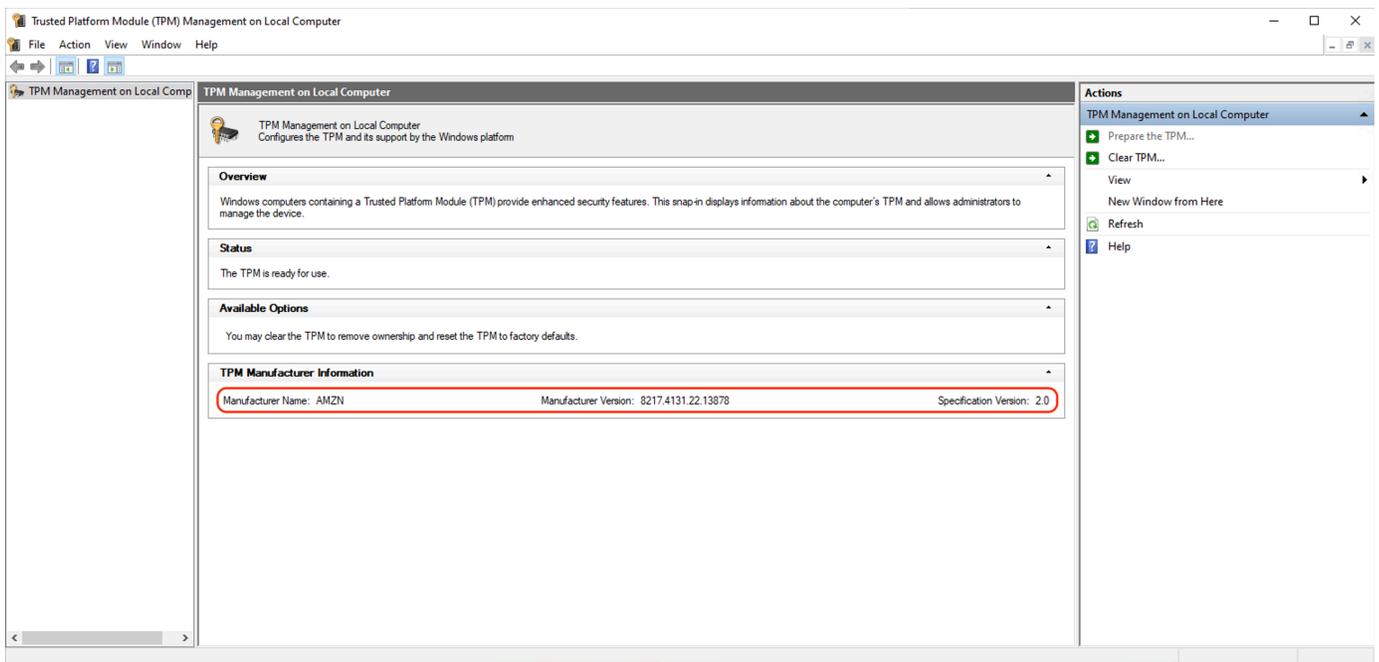
```
"Instances": {  
  "InstanceId": "i-0123456789example",  
  "InstanceType": "c5.large",  
  ...  
  "BootMode": "uefi",  
  "TpmSupport": "v2.0"  
  ...  
}
```

(Instans Windows) Untuk memverifikasi apakah NitroTPM dapat diakses di dalam instans Windows Amazon EC2

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.



Ambil kunci dukungan publik untuk sebuah contoh

Anda dapat dengan aman mengambil kunci dukungan publik untuk sebuah instans kapan saja menggunakan AWS CLI

Untuk mengambil kunci dukungan publik untuk sebuah contoh

Gunakan AWS CLI perintah [get-instance-tpm-ek-pub](#).

Contoh 1

Contoh perintah berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `tpmt` format misalnya `i-01234567890abcdef`

```
$ aws ec2 get-instance-tpm-ek-pub \  
--instance-id i-01234567890abcdef \  
--key-format tpmt \  
--key-type rsa-2048
```

Berikut ini adalah contoh output.

```
{  
  "InstanceId": "i-01234567890abcdef",  
  "KeyFormat": "tpmt",  
  "KeyType": "rsa-2048",  
  "KeyValue": "AAEACwADALIAIINx12dEhLEXAMPLEUa11yT9UtduB1ILZPKh2hszFGmqAAYAgABDA  
EXAMPLEAAABA0iRd7WmgtdGNoV1h/AxmW+CXExblG8pEUfNm0L0LiYnEXAMPLERqApiFa/UhvEYqN4  
Z7jKMD/usbhsQaAB1gKA5RmzuhSazHQkax7EXAMPLEzDth1S7HNGuYn5eG7qnJndRcakS+iNxT8Hvf  
0S1ZtNuItMs+Yp4S06aU28MT/JZk0KsXIdMerY3GdWbNQz9AvYbMEXAMPLEPyHfzgV00QTTJVGdDxh  
vxtXC0u9GYf0crbjEXAMPLEd4YTbWdDdg0KWF9fjzDytJSDhrLAOUctNzHPCd/9215zEXAMPLE0IFA  
Ss50C0/802c17W2pMSVHvCCa91YCiAfxH/vYKovAAE="
```

Contoh 2

Contoh perintah berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `der` format misalnya `i-01234567890abcdef`

```
$ aws ec2 get-instance-tpm-ek-pub \  
--instance-id i-01234567890abcdef \  
--key-format der
```

```
--key-format der \  
--key-type rsa-2048
```

Berikut ini adalah contoh output.

```
{  
  "InstanceId": "i-01234567890abcdef",  
  "KeyFormat": "der",  
  "KeyType": "rsa-2048",  
  "KeyValue": "MIIBIjANBgEXAMPLew0BAQEFAAOCAQ8AMIIBCgKCAQEA6JF3taEXAMPLExWH8DGZb4  
JcTFuUbykRR82bQs4uJifaKS0v5NGoEXAMPLEG8Rio3hnuMowP+6xuGxBoAHWAoD1Gb06FJrMdEXAMP  
LEnYUHvM02GVLsc0a5if14buqcmd1FxqRL6I3FPwe9/REXAMPLE0yz5inhI7ppTbwxP81mQ4qxch0x6  
tjcZ1Zs1DP0EXAMPLERUYLQ/Id/OBU7RBNM1UZ0PGG/G1cI670Zh/Rytu0dx9iEXAMPLEtZ0N2A4pYX  
1+PMPK01I0GssA5Ry03Mc8J3/3aXn0D2/ASRQ4gUBKznQLT/zTZEXAMPLEJUe8IJr2VgKIB/Ef+9gqi  
8AAQIDAQAB"  
}
```

Credential Guard untuk instance 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. Ketika Anda menjalankan instans EC2 Windows, Credential Guard menggunakan Sistem AWS Nitro untuk melindungi kredensial login Windows agar tidak diekstraksi dari memori OS.

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 memanfaatkan Credential Guard:

Amazon Machine Image (AMI)

AMI harus dikonfigurasi sebelumnya untuk mengaktifkan Secure Boot NitroTPM dan UEFI. Untuk informasi selengkapnya tentang AMI yang didukung, lihat [the section called “Prasyarat”](#).

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

Tipe instans berikut mendukung Credential Guard di semua ukuran: C5, C5d, C5n, C6i, C6id, C6in, M5, M5d, M5dn, M5n, M5zn, M6i, M6id, M6idn, M6in, R5, R5b, R5d, R5dn, R5n, R6i, R6id, R6idn, R6in.

Note

Meskipun NitroTPM memiliki beberapa tipe instans wajib yang sama, tipe instans harus salah satu di atas untuk mendukung Credential Guard.

Luncurkan instance yang didukung

Anda dapat menggunakan konsol Amazon EC2 atau AWS Command Line Interface (AWS CLI) untuk meluncurkan instance 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

Amazon EC2 console

Untuk meluncurkan instans menggunakan konsol Amazon EC2

Ikuti langkah-langkah untuk [meluncurkan instance](#), menentukan jenis instans yang didukung dan AMI Windows yang telah dikonfigurasi sebelumnya.

AWS CLI

Untuk meluncurkan instance menggunakan AWS CLI

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-id \  
  --key-name key-name
```

PowerShell

Untuk meluncurkan instance menggunakan AWS Tools for PowerShell

Gunakan perintah [New-EC2Instance](#) untuk meluncurkan instans menggunakan tipe instans yang didukung dan AMI Windows yang telah dikonfigurasi sebelumnya.

```
New-EC2Instance `br/>  -ImageId resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base `br/>  -InstanceType c6i.large `br/>  -Region us-east-1 `br/>  -SubnetId subnet-id `br/>  -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 Virtualization Based Protection of Code Integrity:

- 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 “Connect ke instans Windows Anda 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

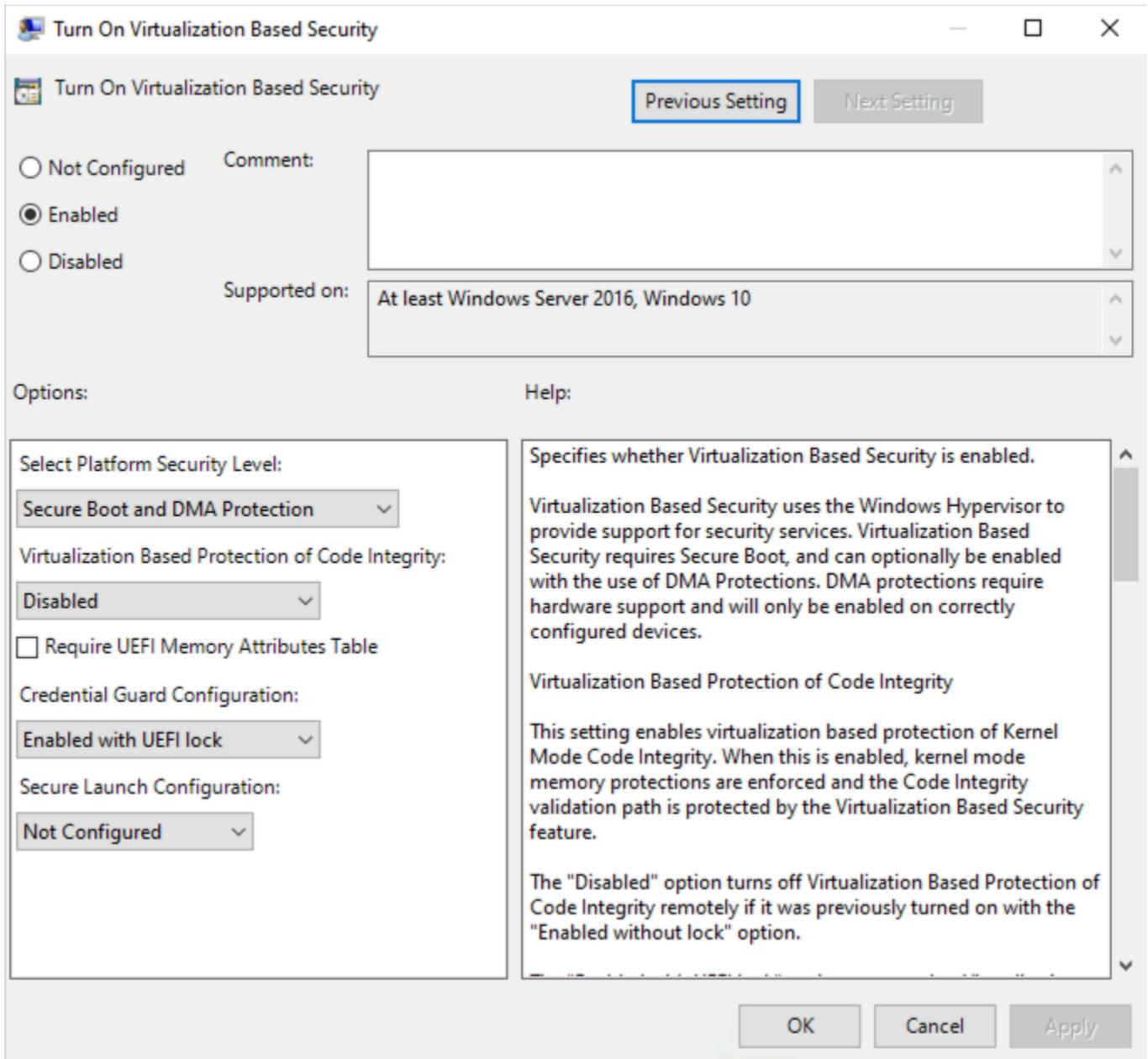
1. Terhubung ke instans Anda sebagai akun pengguna dengan hak akses administrator menggunakan Protokol Desktop Jarak Jauh (RDP). Untuk informasi selengkapnya, lihat [the section called “Connect ke instans Windows Anda menggunakan klien RDP”](#).

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 Konfigurasi Credential Guard, pilih Diaktifkan dengan kunci UEFI.

 Note

Pengaturan kebijakan yang tersisa tidak diperlukan untuk mengaktifkan Credential Guard dan dapat dibiarkan sebagai Tidak Dikonfigurasi.

Gambar berikut menampilkan pengaturan VBS yang dikonfigurasi seperti yang dijelaskan sebelumnya:



9. Boot ulang instans untuk menerapkan pengaturan.

Verifikasi bahwa Credential Guard sedang berjalan

Anda dapat menggunakan alat Informasi Sistem Microsoft (`Msiinfo32.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 “Connect ke instans Windows Anda 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 samping Layanan keamanan berbasis Virtualisasi, konfirmasi bahwa Credential Guard muncul sebagai Berjalan.

Gambar berikut menampilkan VBS berjalan seperti yang dijelaskan sebelumnya:

Virtualization-based security	Running
Virtualization-based security Required Security Properties	Base Virtualization Support, Secure Boot, DMA Protection
Virtualization-based security Available Security Properties	Base Virtualization Support, Secure Boot, DMA Protection, UEFI Code Readonly, Mode Based Execution Control
Virtualization-based security Services Configured	Credential Guard
Virtualization-based security Services Running	Credential Guard

Opsi penyimpanan untuk instans Amazon EC2 Anda

Amazon EC2 memberi Anda opsi penyimpanan easy-to-use data yang fleksibel, hemat biaya, dan 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.

[Amazon EBS](#)

Amazon EBS menyediakan volume penyimpanan tingkat blok yang tahan lama yang dapat Anda pasang 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 Instans](#)

Penyimpanan instans menyediakan penyimpanan tingkat blok sementara untuk instans tertentu. 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.

[Amazon EFS](#)(Hanya contoh Linux)

Amazon EFS menyediakan penyimpanan file yang dapat diskalakan untuk digunakan dengan Amazon EC2. Anda dapat membuat sistem file EFS dan mengonfigurasi 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.

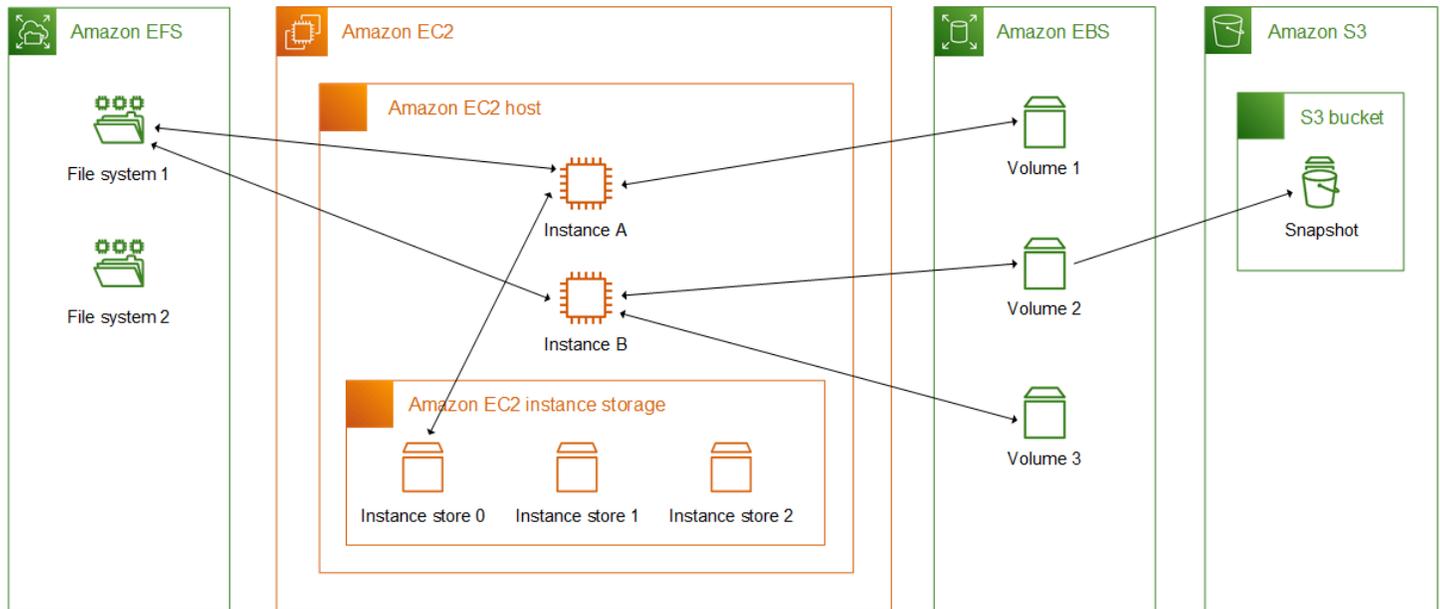
[Amazon S3](#)

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 yang didukung penyimpanan instans.

Amazon FSx

Dengan Amazon FSx, Anda dapat meluncurkan, menjalankan, dan menskalakan sistem file berperforma 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.

Gambar berikut ini menunjukkan hubungan antara opsi penyimpanan ini dan instans Anda.



Harga penyimpanan

Buka [AWS Harga](#), gulir ke Harga untuk AWS produk dan pilih Penyimpanan. Pilih produk penyimpanan untuk membuka halaman harga.

Gunakan Amazon EBS dengan Amazon EC2

Amazon Elastic Block Store (Amazon EBS) menyediakan sumber daya penyimpanan blok berkinerja tinggi yang dapat diskalakan yang dapat digunakan dengan instans Amazon Elastic Compute Cloud (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 sebuah 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 point-in-time cadangan volume Amazon EBS yang bertahan secara independen dari volume itu sendiri. Anda dapat membuat snapshot untuk mencadangkan data pada volume Amazon EBS Anda. Anda kemudian dapat memulihkan volume baru dari snapshot tersebut kapan saja.

Anda dapat membuat dan melampirkan volume Amazon EBS ke instans selama peluncuran, dan Anda dapat membuat dan melampirkan volume EBS ke instans kapan saja setelah peluncuran. Dan Anda dapat membuat snapshot dari volume kapan saja setelah pembuatan.

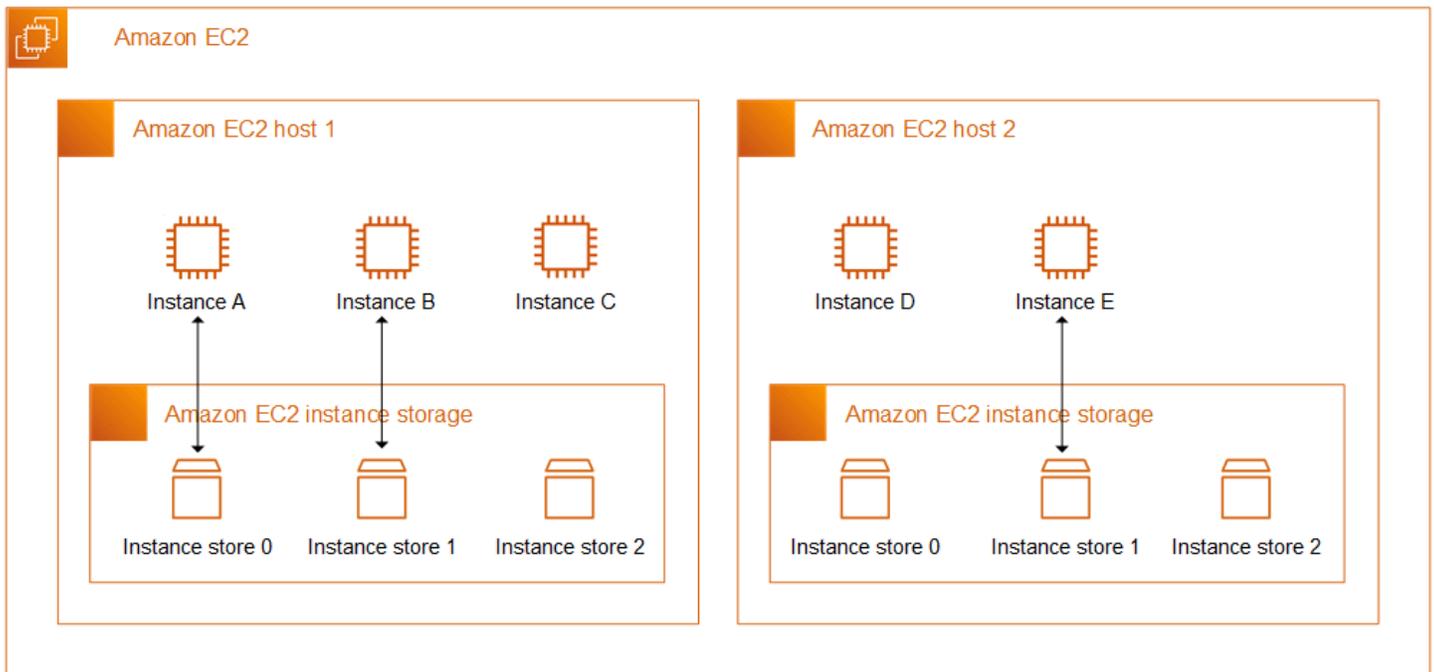
Untuk informasi selengkapnya tentang bekerja dengan volume dan snapshot, lihat [Panduan Pengguna Amazon EBS](#).

Penyimpanan instans Amazon EC2

Penyimpanan instans menyediakan penyimpanan tingkat blok sementara untuk instans Anda. Penyimpanan ini terletak pada disk yang secara fisik terpasang pada 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. Untuk informasi selengkapnya, lihat [Volume penyimpanan instans](#).

Perangkat virtual untuk volume penyimpanan instans adalah `ephemeral[0-23]`. Tipe instans yang mendukung satu volume penyimpanan instans memiliki `ephemeral0`. Tipe instans yang mendukung dua atau lebih volume penyimpanan instans `ephemeral0ephemeral1`, dan sebagainya.



Harga penyimpanan instans

Volume penyimpanan instans disertakan sebagai bagian dari biaya penggunaan.

Daftar Isi

- [Volume penyimpanan instans dan masa pakai data](#)
- [Volume penyimpanan instans](#)
- [Tambahkan volume penyimpanan instans ke instans EC2 Anda](#)
- [Volume penyimpanan instans SSD](#)
- [Volume swap penyimpanan instans untuk instance Linux](#)
- [Optimalkan kinerja disk untuk volume penyimpanan instance pada instance Linux](#)

Volume penyimpanan instans dan masa pakai data

Jumlah, ukuran, dan tipe volume penyimpanan instans ditentukan oleh tipe instans dan ukuran instans. Untuk informasi selengkapnya, lihat [Volume penyimpanan instans](#).

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 diakhiri. 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?
Peristiwa siklus hidup instans yang diinisiasi pengguna	
Instance di-boot ulang	Data tetap ada
Instance dihentikan	Data tidak bertahan
Contohnya hibernasi	Data tidak bertahan
Instance dihentikan	Data tidak bertahan
Jenis instance diubah	Data tidak bertahan *
AMI yang didukung EBS dibuat dari instance	Data tidak bertahan di AMI yang dibuat**
Sebuah instance store-backed AMI dibuat dari instance (instance Linux)	Data tetap ada dalam bundel AMI yang diunggah ke Amazon S3 ***
Peristiwa OS yang diinisiasi pengguna	
Shutdown dimulai	Data tidak bertahan †

Peristiwa	Apa yang terjadi pada data Anda?
Restart dimulai	Data tetap ada
AWS acara terjadwal	
Contoh berhenti	Data tidak bertahan
Contoh reboot	Data tetap ada
Reboot sistem	Data tetap ada
Pensiun contoh	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
Kegagalan daya	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 AMI yang didukung EBS, 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.

Volume penyimpanan instans

Jumlah, ukuran, dan tipe volume penyimpanan instans ditentukan oleh tipe instans dan ukuran instans. Beberapa tipe instans, seperti M6, C6, dan R6, tidak mendukung volume penyimpanan instans, sementara tipe instans lainnya, seperti M5d, C6gd, dan R6gd, 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, `m5d.large` mendukung volume penyimpanan instans 1 x 75 GB, sementara `m5d.24xlarge` mendukung volume penyimpanan instans 4 x 900 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 tipe instans menggunakan NVMe atau SSD berbasis SATA (SSD), sementara yang lain menggunakan hard disk drive berbasis SATA (HDD). SSD menghasilkan performa I/O acak tinggi dengan latensi 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](#).

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 pengoptimalan kuantitas, ukuran, jenis, dan kinerja volume penyimpanan instans yang tersedia pada setiap jenis instans yang didukung. Untuk informasi selengkapnya, lihat berikut ini:

- [Spesifikasi toko instans — Tujuan umum](#)
- [Spesifikasi toko instans — Komputasi dioptimalkan](#)
- [Spesifikasi toko instans - Memori dioptimalkan](#)
- [Spesifikasi toko instans — Penyimpanan dioptimalkan](#)

- [Spesifikasi toko instans — Komputasi yang dipercepat](#)
- [Spesifikasi toko instans — Komputasi berkinerja tinggi](#)
- [Spesifikasi toko instans — Generasi sebelumnya](#)

Untuk mengambil informasi volume penyimpanan instance menggunakan AWS CLI

Anda dapat menggunakan AWS CLI perintah [describe-instance-types](#) untuk menampilkan informasi tentang jenis instance, seperti volume penyimpanan instance-nya. Contoh berikut menampilkan ukuran total penyimpanan instans untuk semua instans R5 dengan volume penyimpanan instan.

```
aws ec2 describe-instance-types \
  --filters "Name=instance-type,Values=r5*" "Name=instance-storage-
supported,Values=true" \
  --query "InstanceTypes[][InstanceType, InstanceStorageInfo.TotalSizeInGB]" \
  --output table
```

Contoh Output

```
-----
| DescribeInstanceTypes |
+-----+-----+
| r5ad.24xlarge | 3600 |
| r5ad.12xlarge | 1800 |
| r5dn.8xlarge  | 1200 |
| r5ad.8xlarge  | 1200 |
| r5ad.large    | 75   |
| r5d.4xlarge   | 600  |
| . . .        |      |
| r5dn.2xlarge  | 300  |
| r5d.12xlarge  | 1800 |
+-----+-----+
```

Contoh berikut menampilkan detail penyimpanan instans lengkap untuk tipe instans yang ditentukan.

```
aws ec2 describe-instance-types \
  --filters "Name=instance-type,Values=r5d.4xlarge" \
  --query "InstanceTypes[].InstanceStorageInfo"
```

Contoh output menunjukkan bahwa tipe instans ini memiliki dua volume SSD NVMe 300 GB, untuk total 600 GB penyimpanan instans.

```
[
  {
    "TotalSizeInGB": 600,
    "Disks": [
      {
        "SizeInGB": 300,
        "Count": 2,
        "Type": "ssd"
      }
    ],
    "NvmeSupport": "required"
  }
]
```

Tambahkan volume penyimpanan instans ke instans EC2 Anda

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 instance 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 [Pemetaan perangkat blok](#)

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 suatu instans yang didukung penyimpanan instans akan dipasang secara otomatis.

Pertimbangan untuk volume root

Pemetaan perangkat blok selalu menentukan volume root untuk instans tersebut. Volume root selalu dipasang secara otomatis.

Instance 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 [Penyimpanan untuk perangkat 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](#)
- [Menambahkan volume penyimpanan instans non-NVME ke instans Anda](#)
- [Sediakan volume penyimpanan instans di instans Anda](#)

Menambahkan volume penyimpanan instans ke AMI

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.

Pertimbangan

- 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.

- Saat Anda meluncurkan instans, Anda dapat menghilangkan volume penyimpanan instans non-NVMe yang ditentukan dalam pemetaan perangkat blok AMI atau menambahkan volume penyimpanan instans.

New console

Untuk menambahkan volume penyimpanan instans ke AMI yang didukung Amazon EBS menggunakan konsol

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. (Untuk informasi selengkapnya, lihat [Nama perangkat di instans Amazon EC2](#).) Jumlah volume penyimpanan instans yang tersedia bergantung pada tipe instans. Untuk instans dengan volume penyimpanan instans NVMe, pemetaan perangkat volume ini bergantung pada urutan sistem operasi yang melakukan enumerasi volume tersebut.
6. Pilih Buat citra.

AWS CLI

Untuk menambahkan volume penyimpanan instans ke AMI menggunakan baris perintah

Anda dapat menggunakan salah satu dari perintah berikut. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [create-image](#) atau [register-image](#) (AWS CLI)
- [New-EC2Image](#) dan [Register-EC2Image](#) AWS Tools for Windows PowerShell

Menambahkan volume penyimpanan instans non-NVME ke instans Anda

Saat Anda meluncurkan instans yang mendukung volume penyimpanan instans non-NVMe, Anda harus menentukan pemetaan perangkat blok agar volume penyimpanan instans dapat dilampirkan. 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.

Pertimbangan

- Untuk instans M3, Anda dapat menerima volume penyimpanan instans bahkan jika Anda tidak menentukannya dalam pemetaan perangkat blok untuk instans tersebut.

Untuk menentukan pemetaan perangkat blok dalam permintaan peluncuran instans, gunakan salah satu metode berikut.

Amazon EC2 console

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 Konfigurasi penyimpanan, bagian Volume penyimpanan Instans mencantumkan volume penyimpanan instans yang dapat dilampirkan ke instans. Jumlah volume penyimpanan instans yang tersedia bergantung pada tipe instans.
5. Untuk setiap volume penyimpanan instans yang akan dilampirkan, untuk Nama perangkat, pilih nama perangkat yang akan digunakan.
6. Konfigurasi pengaturan instans yang tersisa sesuai kebutuhan, lalu pilih Luncurkan instans.

Command line

Anda dapat menggunakan salah satu perintah opsi berikut dengan opsi yang sesuai.

- `--block-device-mappings` dengan [run-instans](#) (AWS CLI)
- `-BlockDeviceMapping` dengan [New-EC2Instance](#) (AWS Tools for Windows PowerShell)

Sediakan volume penyimpanan instans di instans Anda

Setelah meluncurkan instans dengan volume penyimpanan instans terlampir, Anda harus memasang volume sebelum dapat mengaksesnya.

Note

Banyak volume penyimpanan instans sudah diformat sebelumnya untuk sistem file ext3. Volume penyimpanan instans berbasis SSD yang mendukung instruksi TRIM tidak diformat sebelumnya untuk setiap sistem file. Namun, Anda dapat memformat volume dengan sistem file pilihan Anda setelah Anda meluncurkan instans. Untuk informasi selengkapnya, lihat [Dukungan TRIM volume penyimpanan instans](#). Untuk instans Windows, kami memformat ulang volume penyimpanan instans dengan sistem file NTFS.

Instans Linux

Anda dapat melihat dan memasang volume penyimpanan instance seperti yang dijelaskan dalam prosedur berikut.

Untuk membuat volume penyimpanan instans tersedia di Linux

1. Sambungkan ke instans Anda menggunakan SSH. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Linux Anda](#).
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

Anda juga dapat melihat volume penyimpanan instance menggunakan Windows Disk Management. Untuk informasi selengkapnya, lihat [Mencantumkan disk menggunakan Manajemen Disk](#).

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 inialisasi, pilih jenis partisi yang diperlukan bergantung pada kasus penggunaan Anda, lalu pilih OKE.
4. Pada daftar volume, klik kanan volume yang akan dipasang, kemudian pilih Volume Sederhana Baru.
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 atau tidak. Jika Anda memilih untuk memformat volume, pilih sistem file dan ukuran unit yang diperlukan, dan tentukan label volume.
 9. Pilih Selanjutnya, Selesai.

Untuk petunjuk tentang cara memasang volume yang terpasang secara otomatis setelah reboot, lihat [Memasang volume yang terpasang secara otomatis setelah reboot](#) di Panduan Pengguna Amazon EBS.

Volume penyimpanan instans SSD

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 [Tambahkan volume penyimpanan instans ke instans EC2 Anda](#).

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 [Volume penyimpanan instans](#).

Data pada penyimpanan instans NVMe dienkripsi menggunakan cipher blok XTS-AES-256 yang diimplementasikan pada modul perangkat keras di instans tersebut. 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 Jenis instance berbasis graviton memerlukan Ubuntu 18.04 atau 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 setelahnya
- 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.xlarge`, 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, SUSE Linux Enterprise Server, 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/nvme/nvme-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 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 AWS NVMe. Jika Anda tidak menggunakan AMI AWS Windows terbaru, Anda dapat [menginstal driver AWS NVMe saat ini](#).

Volume SSD Non-NVMe

Instans berikut mendukung volume penyimpanan instans yang menggunakan SSD non-NVMe untuk memberikan kinerja I/O acak yang tinggi: C3, I2, M3, R3, dan X1. Untuk informasi selengkapnya tentang volume penyimpanan instans yang didukung oleh setiap tipe instans, lihat [Volume penyimpanan instans](#).

Performa I/O volume penyimpanan instans berbasis SSD

Saat Anda mengisi volume penyimpanan instans berbasis SSD untuk instans Anda, jumlah IOPS tulis yang dapat Anda capai akan menurun. 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 performa I/O yang drastis.

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. Volume penyimpanan instans berbasis SSD yang disediakan untuk sebuah instans tidak memiliki ruang yang disediakan untuk penyediaan berlebih. Untuk mengurangi amplifikasi tulis, sebaiknya Anda membiarkan 10 persen volume tidak dipartisi sehingga pengontrol SSD dapat menggunakannya untuk penyediaan berlebih. 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 [Volume penyimpanan instans](#).

Note

(Hanya instance Windows) Instans yang menjalankan Windows Server 2012 R2 mendukung TRIM pada AWS PV Driver 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 perintah `fsutil behavior set DisableDeleteNotify 1` 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.

Volume swap penyimpanan instans untuk instance Linux

Note

Topik ini hanya berlaku untuk instance 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 [Ubah tipe instans](#).

Tipe instans `c1.medium` dan `m1.small` memiliki jumlah memori fisik yang terbatas untuk digunakan, dan diberi volume swap 900 MiB pada saat peluncuran untuk bertindak sebagai memori virtual untuk Linux AMI. Meskipun kernel Linux melihat ruang swap ini sebagai partisi pada perangkat root, sebenarnya ini adalah volume penyimpanan instans terpisah, terlepas dari jenis perangkat 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 [Volume penyimpanan instans](#).

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.

Optimalkan kinerja disk untuk volume penyimpanan instance pada instance Linux

Note

Topik ini hanya berlaku untuk instance Linux.

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

Beberapa tipe instans dengan solid state drive (SSD) pasang langsung dan dukungan TRIM memberikan performa maksimal saat peluncuran, tanpa inisialisasi. Untuk informasi tentang penyimpanan instans untuk setiap tipe instans, lihat [Volume penyimpanan instans](#).

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 RAID dengan melakukan ke 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
```

Penyimpanan file

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 instans Amazon EC2.

Daftar Isi

- [Menggunakan Amazon S3 dengan Amazon EC2](#)
- [Menggunakan Amazon EFS dengan instans Linux](#)
- [Menggunakan Amazon FSx dengan Amazon EC2](#)
- [Gunakan Cache File Amazon dengan Amazon EC2](#)

Menggunakan Amazon S3 dengan 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, backup, dan analisis data besar, dari instans Amazon EC2 atau dari mana saja melalui internet. Untuk informasi selengkapnya, lihat [Apa itu Amazon S3?](#)

Objek adalah entitas dasar yang disimpan di Amazon S3. Setiap objek yang disimpan di Amazon S3 dimuat dalam bucket. Bucket atau GA namespace Amazon S3 di tingkat tertinggi dan identifikasi akun yang bertanggung jawab atas penyimpanan tersebut. Bucket Amazon S3 mirip dengan nama domain internet. Objek yang disimpan di dalam bucket memiliki nilai kunci yang unik dan diambil menggunakan URL. Sebagai contoh, jika sebuah objek dengan nilai kunci /photos/mygarden.jpg disimpan di dalam bucket DOC-EXAMPLE-BUCKET1, objek tersebut dapat dialamatkan menggunakan URL `https://DOC-EXAMPLE-BUCKET1.s3.amazonaws.com/photos/mygarden.jpg`. Untuk informasi selengkapnya, lihat [Cara kerja Amazon S3](#).

Contoh penggunaan

Mengingat keuntungan Amazon S3 untuk penyimpanan, Anda mungkin memutuskan untuk menggunakan layanan ini dalam menyimpan file dan set data untuk digunakan dengan instans EC2. Ada berbagai cara untuk memindahkan data ke dan dari Amazon S3 ke instans Anda. Selain contoh-contoh yang dibahas di bawah ini, ada berbagai alat yang telah ditulis orang yang dapat Anda gunakan untuk mengakses data Anda di Amazon S3 dari komputer atau instans Anda. Beberapa hal yang umum dibahas di forum AWS .

Jika Anda memiliki izin, Anda dapat menyalin file ke atau dari Amazon S3 dan instans Anda menggunakan salah satu metode berikut.

GET or wget (Linux)

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 [Manajemen identitas dan akses di Amazon S3](#) dan [Mengunduh objek](#) di Panduan Pengguna Amazon S3.

Utilitas wget adalah klien HTTP dan FTP yang memungkinkan Anda 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://my_bucket.s3.amazonaws.com/path-to-file
```

AWS Tools for Windows PowerShell (Windows)

Instans Windows memiliki keunggulan berupa peramban grafis yang bisa Anda gunakan untuk mengakses konsol Amazon S3 secara langsung; namun, untuk keperluan skrip, pengguna Windows juga bisa menggunakan [AWS Tools for Windows PowerShell](#) untuk memindahkan objek ke dan dari Amazon S3.

Gunakan perintah berikut untuk menyalin objek Amazon S3 ke instans Windows Anda.

```
PS C:\> Copy-S3Object -BucketName my_bucket -Key path-to-file -  
LocalFile my_copied_file.ext
```

AWS CLI (Linux and Windows)

The AWS Command Line Interface (AWS CLI) adalah alat terpadu untuk mengelola AWS layanan Anda. AWS CLI memungkinkan pengguna mengautentikasi sendiri dan mengunduh item terbatas dari Amazon S3 serta mengunggah item. Untuk informasi selengkapnya, seperti cara menginstal dan mengonfigurasi alat, lihat [halaman detail AWS Command Line Interface](#).

Perintah `aws s3 cp` mirip dengan perintah `cp` 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 instans Anda.

```
aws s3 cp s3://my_bucket/my_folder/my_file.ext my_copied_file.ext
```

Gunakan perintah berikut untuk menyalin objek dari instans Anda kembali ke Amazon S3.

```
aws s3 cp my_copied_file.ext s3://my_bucket/my_folder/my_file.ext
```

Perintah `aws s3 sync` dapat menyinkronkan seluruh bucket Amazon S3 dengan lokasi direktori lokal. Ini dapat membantu untuk mengunduh kumpulan data dan menyimpan salinan lokal up-to-date 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 instans Anda.

```
aws s3 sync s3://remote_S3_bucket local_directory
```

Amazon S3 API

Jika Anda adalah developer, 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 Panduan Pengguna Amazon S3](#).

Menggunakan Amazon EFS dengan instans 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 Amazon EFS Quick Create 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.
- [Pasang target di setiap subnet default](#) di VPC yang dipilih.
- [Mode kinerja Tujuan Umum](#).
- Mode [throughput 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 Konfigurasi 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 titik Mount, tentukan titik pemasangan khusus atau pertahankan default.
 - 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 — Termasuk aturan keluar yang memungkinkan lalu lintas melalui port NFS 2049, 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 ditimpa oleh data pengguna yang dihasilkan secara otomatis.

9. Konfigurasi pengaturan instans lain sesuai kebutuhan.
10. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#).

Menguji sistem file EFS

Anda dapat terhubung ke instans dan memverifikasi bahwa sistem file terpasang pada 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](#).
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 ini.

```
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.

Menggunakan Amazon FSx dengan 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 [Meluncurkan sebuah instans menggunakan wizard peluncuran instans baru](#)
- 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	<i>grup keamanan sistem file</i>
UDP	20001 - 20003	<i>grup keamanan sistem file</i>
UDP	4049	<i>grup keamanan sistem file</i>
UDP	2049	<i>grup keamanan sistem file</i>
UDP	635	<i>grup keamanan sistem file</i>
UDP	4045 - 4046	<i>grup keamanan sistem file</i>
TCP	4049	<i>grup keamanan sistem file</i>
TCP	635	<i>grup keamanan sistem file</i>
TCP	2049	<i>grup keamanan sistem file</i>
TCP	111	<i>grup keamanan sistem file</i>
TCP	4045 - 4046	<i>grup keamanan sistem file</i>
TCP	20001 - 20003	<i>grup keamanan sistem file</i>

Tipe protokol	Nomor port	Tujuan
Semua	Semua	<i>grup keamanan sistem file</i>

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	<i>grup keamanan instans</i>
UDP	20001 - 20003	<i>grup keamanan instans</i>
UDP	4049	<i>grup keamanan instans</i>
UDP	111	<i>grup keamanan instans</i>
UDP	635	<i>grup keamanan instans</i>
UDP	4045 - 4046	<i>grup keamanan instans</i>
TCP	4045 - 4046	<i>grup keamanan instans</i>
TCP	635	<i>grup keamanan instans</i>
TCP	2049	<i>grup keamanan instans</i>
TCP	4049	<i>grup keamanan instans</i>
TCP	20001 - 20003	<i>grup keamanan instans</i>
TCP	111	<i>grup keamanan instans</i>

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} ${fsx_mount_point_1} nfs4
nfsvers=4.1,rsz=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev
0 0\n" >> /etc/fstab; else printf "\n${svm_id_1}.${file_system_id_1}.fsx.eu-
north-1.amazonaws.com/${vol_path_1} ${fsx_mount_point_1} nfs4
nfsvers=4.1,rsz=1048576,wsz=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. Konfigurasi 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 Amazon FSx untuk NetApp ONTAP dan Amazon FSx untuk sistem file 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 ingin 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 ingin memberikan skrip data pengguna secara manual, kosongkan kotak centang. Untuk informasi selengkapnya, lihat [Skrip data pengguna](#).
8. Di bagian Lanjutan, konfigurasi pengaturan instans tambahan sesuai kebutuhan.
9. Pilih Luncurkan.

Gunakan Cache File Amazon dengan Amazon EC2

Amazon File Cache adalah cache berkecepatan tinggi AWS yang dikelola sepenuhnya yang digunakan 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 [Apa itu Cache File Amazon?](#)

Anda dapat mengakses cache dari instans Amazon EC2 menggunakan klien Lustre open-source. Instans Amazon EC2 dapat mengakses cache Anda dari Availability Zone lain dalam Amazon Virtual Private Cloud (Amazon VPC) yang sama, asalkan jaringan Anda memungkinkan akses di seluruh subnet dalam VPC. Setelah cache Anda dipasang, Anda dapat bekerja dengan file dan direktorinya seperti yang Anda lakukan saat menggunakan sistem file lokal.

Untuk memulai, lihat [Memulai dengan Cache File Amazon](#).

Batasan volume instans

Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Ketika mempertimbangkan berapa banyak volume yang akan dilampirkan ke instans Anda, Anda harus mempertimbangkan apakah Anda membutuhkan bandwidth I/O yang lebih besar atau kapasitas penyimpanan yang lebih besar.

Bandwidth versus kapasitas

Untuk kasus penggunaan bandwidth yang konsisten dan dapat diprediksi, gunakan instans Amazon EBS yang dioptimalkan 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 peningkatan performa karena peningkatan I/O overhead. Uji performa aplikasi individual Anda dan sesuaikan kebutuhan.

Topik

- [Batas volume untuk instans yang dibangun di atas Sistem Nitro](#)
- [Batas volume untuk instans berbasis Xen](#)

Batas volume untuk instans yang dibangun di atas Sistem Nitro

Topik

- [Batas volume Amazon EBS khusus](#)
- [Batas volume Amazon EBS bersama](#)

Batas volume Amazon EBS khusus

Jenis instans Nitro berikut memiliki batas volume Amazon EBS khusus yang bervariasi tergantung pada ukuran instans. Batas tidak dibagikan dengan lampiran perangkat lain. Dengan kata lain, Anda dapat melampirkan sejumlah volume Amazon EBS hingga batas pelampiran volume, berapa pun jumlah perangkat yang dilampirkan, seperti volume penyimpanan instans NVMe dan antarmuka jaringan.

- Tujuan umum: M7a, M7i, M7i-flex
- Komputasi yang dioptimalkan: C7a, C7i
- Memori yang dioptimalkan: R7a, R7i, R7iz

Untuk jenis instance yang mendukung batas volume khusus ini, batas volume bergantung pada ukuran instans. Tabel berikut menunjukkan batas untuk setiap ukuran instans.

Ukuran instans	Batas Volume
medium large xlarge 2xlarge 4xlarge 8xlarge 12xlarge	32
16xlarge	48
24xlarge	64
32xlarge	88
48xlarge	128
metal-16x1 metal-24x 1	39
metal-32x1 metal-48x 1	79

Batas volume Amazon EBS bersama

Semua jenis instans Nitro lainnya (tidak tercantum dalam [Batas volume Amazon 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 ini mendukung maksimal 28 lampiran. Misalnya, jika Anda tidak memiliki lampiran antarmuka jaringan tambahan pada suatu instans m5.xlarge, Anda dapat memasang hingga 27 volume EBS (batas volume 28 - 1 antarmuka jaringan). Jika Anda memiliki dua antarmuka jaringan tambahan pada sebuah instans m5.xlarge, Anda dapat melampirkan hingga 25 volume EBS (batas volume 28 - 3 antarmuka jaringan). Demikian pula, jika Anda memiliki dua antarmuka jaringan tambahan pada sebuah instans m5d.xlarge, yang memiliki 1 volume penyimpanan instans NVMe, Anda dapat melampirkan hingga 24 volume EBS (batas volume 28 - 3 antarmuka jaringan - 1 volume penyimpanan instans NVMe).

Pengecualian berikut untuk tipe instance yang memiliki batas volume bersama:

- Instans DL2q mendukung maksimum 19 volume EBS.
- Sebagian besar instans bare metal mendukung maksimal 31 volume EBS.
- Instans tervirtualisasi memori tinggi mendukung maksimum 27 volume EBS.
- Instans bare metal memori tinggi mendukung maksimum 19 volume EBS.
- Instans `inf1.xlarge` dan `inf1.2xlarge` mendukung maksimal 26 volume EBS.
- Instans `inf1.6xlarge` mendukung diberikan maksimal 23 volume EBS.
- Instans `mac1.metal` mendukung diberikan maksimal 16 volume EBS.
- `mac2.metal`, `mac2-m2.metal`, dan `mac2-m2pro.metal` instans mendukung maksimal 10 volume EBS.
- Instans `inf1.24xlarge` mendukung diberikan maksimal 11 volume EBS.
- Instans `g5.48xlarge` mendukung maksimal 9 volume EBS.
- Instans `d3.8xlarge` dan `d3en.12xlarge` mendukung maksimal 3 volume EBS.
- Untuk instans komputasi terakselerasi, akselerator yang terpasang dihitung terhadap batas volume bersama. Misalnya, untuk instans `p4d.24xlarge`, yang memiliki batas volume bersama 28, 8 GPU, dan 8 volume penyimpanan instans NVMe, Anda dapat melampirkan hingga 11 volume Amazon EBS (batas volume 28 - 1 antarmuka jaringan - 8 GPU - 8 volume penyimpanan instans NVMe).

Batas volume untuk instans berbasis Xen

Instans Linux

Melampirkan lebih dari 40 volume ke instans Linux berbasis Xen 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

Memasang lebih dari 40 volume ke instans Linux berbasis Xen hanya didukung dengan upaya terbaik dan tidak dijamin.

Instans Windows

Tabel berikut menunjukkan batas volume untuk instans Windows berbasis Xen berdasarkan driver yang digunakan. Angka-angka ini termasuk volume root, ditambah volume penyimpanan instans dan volume Amazon EBS yang terlampir.

⚠ Important

Melampirkan lebih dari jumlah volume berikut ke instans Windows berbasis Xen hanya didukung berdasarkan upaya terbaik dan tidak dijamin.

Driver	Batas Volume
AWS PV	26
Citrix PV	26
Red Hat PV	17

Kami menyarankan Anda untuk tidak melampirkan lebih dari 26 volume ke instance Windows berbasis Xen 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”](#).

Untuk informasi selengkapnya tentang cara nama perangkat terkait volume, lihat [Petakan disk ke volume pada instans Windows](#).

Volume root instans Amazon EC2

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 Anda selama atau setelah peluncuran.

Kami mencadangkan nama perangkat khusus untuk volume root. Untuk informasi selengkapnya, lihat [Nama perangkat di instans Amazon EC2](#).

Daftar Isi

- [Tipe volume root](#)
- [Pilih AMI Linux berdasarkan jenis volume root](#)
- [Tentukan jenis perangkat root dari instance Linux Anda](#)
- [Ubah volume root agar tetap ada](#)
- [Ubah ukuran awal volume root](#)
- [Ganti volume root instans EC2](#)

Tipe volume root

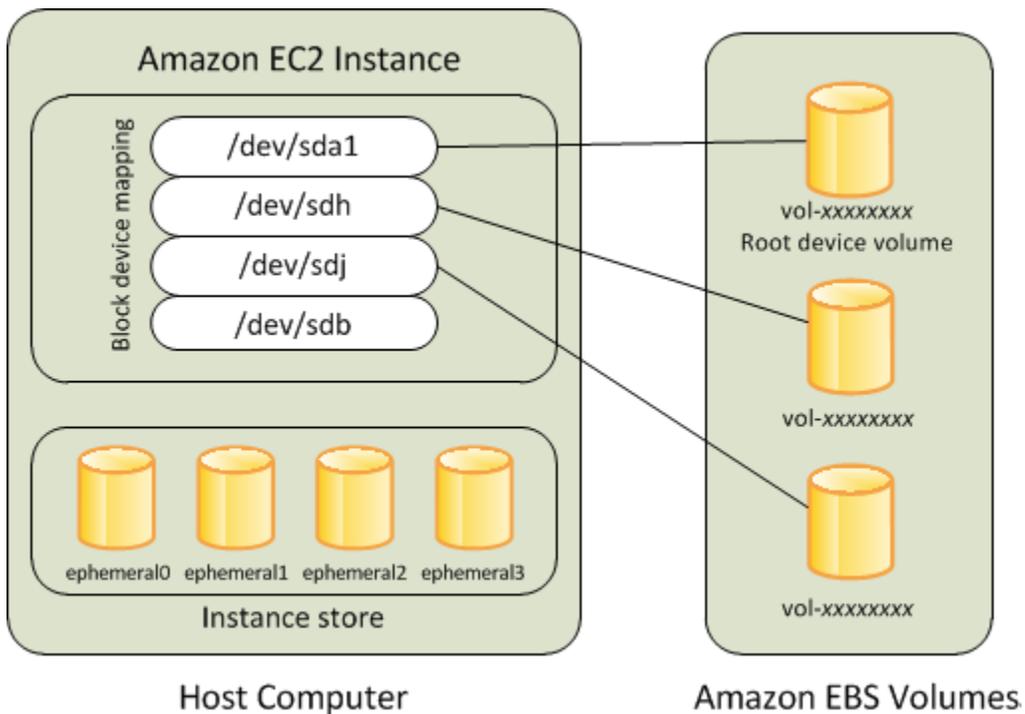
AMI yang Anda gunakan untuk meluncurkan instance menentukan jenis volume root. Anda dapat meluncurkan instance dari AMI yang didukung Amazon EBS-backed (instans Linux dan Windows) atau AMI yang didukung oleh instance store-backed (hanya instans Linux). Ada perbedaan signifikan antara apa yang dapat Anda lakukan dengan setiap jenis AMI. Untuk informasi selengkapnya tentang metrik ini, lihat [Penyimpanan untuk perangkat root](#).

Sebaiknya gunakan AMI yang didukung oleh Amazon EBS, karena instans ini diluncurkan lebih cepat dan menggunakan penyimpanan persisten.

Instans yang didukung Amazon EBS

Instans yang menggunakan Amazon EBS untuk volume root secara otomatis memiliki volume Amazon EBS yang terlampir. Saat Anda meluncurkan instans yang didukung Amazon EBS, kami membuat volume Amazon EBS untuk setiap snapshot Amazon EBS yang dirujuk oleh AMI yang Anda gunakan. Anda dapat menggunakan volume Amazon EBS lain atau volume penyimpanan instans, bergantung pada tipe instans.

Instans yang didukung Amazon EBS dapat dihentikan dan kemudian dimulai ulang tanpa memengaruhi data yang disimpan dalam volume yang dilampirkan. Ada berbagai tugas terkait instans dan volume yang bisa Anda lakukan saat instans yang didukung Amazon EBS dalam status berhenti. Misalnya, Anda dapat memodifikasi properti dari suatu instans, mengubah ukurannya, atau memperbarui kernel yang digunakannya, atau Anda dapat melampirkan volume root ke instans lain yang sedang berjalan untuk melakukan debug atau tujuan lainnya. Untuk informasi selengkapnya, lihat [volume Amazon EBS](#).



Batasan

Anda tidak dapat menggunakan volume EBS st1 atau sc1 sebagai volume root.

Kegagalan instans

Jika instans yang didukung Amazon EBS gagal, Anda dapat memulihkan sesi Anda 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 AMI yang didukung Amazon EBS-Backed](#).
- 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.

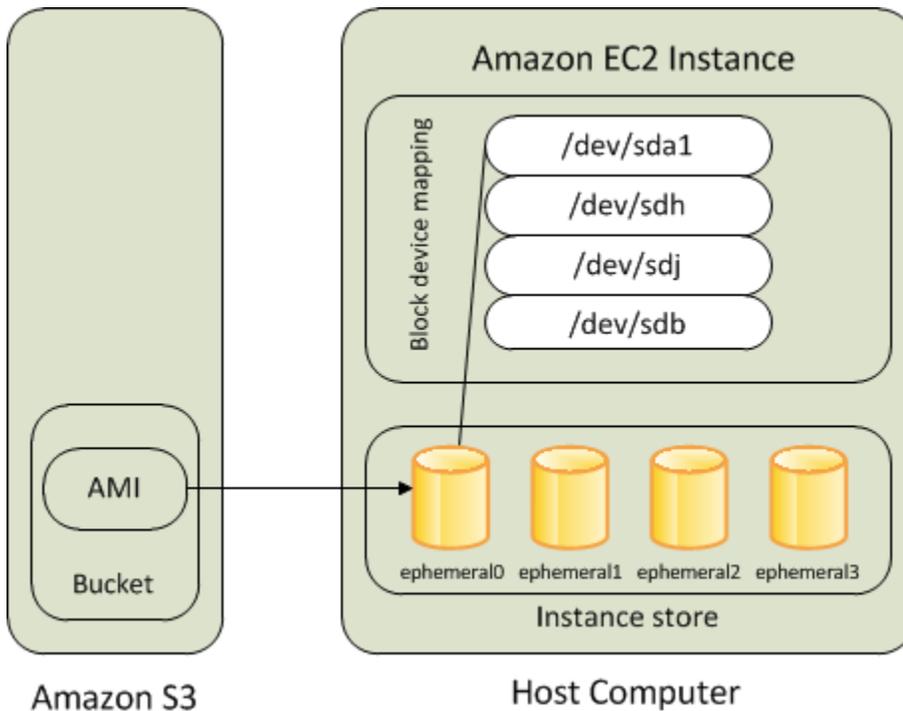
Instans yang didukung toko instans (hanya instance Linux)

Note

Instans Windows tidak mendukung volume root yang didukung instance-store.

Instans yang menggunakan penyimpanan instans untuk volume root secara otomatis memiliki satu atau lebih volume penyimpanan instans yang tersedia, dengan satu volume yang berfungsi sebagai volume root. Saat sebuah instans diluncurkan, gambar yang digunakan untuk booting instans tersebut akan 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 ketika instans dihentikan (instans yang didukung penyimpanan instans tidak mendukung tindakan Hentikan) atau jika gagal (seperti jika drive yang mendasari memiliki masalah). Untuk informasi selengkapnya, lihat [Penyimpanan instans Amazon EC2](#).



Persyaratan

Hanya jenis instance berikut yang mendukung volume penyimpanan instance sebagai volume root: C3, D2, I2, M3, dan R3.

Kegagalan Instans

Setelah instans yang didukung penyimpanan instans gagal atau diakhiri, itu tidak dapat dipulihkan. Jika Anda berencana menggunakan instans yang didukung penyimpanan instans Amazon EC2, kami sangat menyarankan agar Anda mendistribusikan data pada penyimpanan instans Anda di banyak Zona Ketersediaan. Anda juga harus mencadangkan data penting dari volume penyimpanan instans Anda ke penyimpanan persisten secara teratur.

Pilih AMI Linux berdasarkan jenis volume root

Note

Semua AMI Windows didukung EBS.

AMI yang Anda tentukan saat meluncurkan instans menentukan tipe volume perangkat root yang dimiliki instans. Anda dapat melihat AMIs berdasarkan jenis root device menggunakan salah satu metode berikut.

Console

Untuk memilih AMI yang didukung Amazon EBS menggunakan konsol

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih AMI.
3. Dari daftar filter, pilih jenis citra (seperti Citra publik). Di bilah pencarian, pilih Platform untuk memilih sistem operasi (seperti Amazon Linux) dan Root devicetype untuk memilih jenis volume root (ebs atau instance-store).
4. (Opsional) Untuk mendapatkan informasi tambahan untuk membantu Anda membuat pilihan, pilih ikon Preferensi, alihkan kolom untuk ditampilkan, lalu pilih Konfirmasi.
5. Pilih AMI dan tulis ID AMI-nya.

AWS CLI

Untuk memverifikasi tipe volume perangkat root dari AMI menggunakan baris perintah

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-images](#) (AWS CLI)
- [Get-EC2Image](#) (AWS Tools for Windows PowerShell)

Tentukan jenis perangkat root dari instance Linux Anda

Note

Semua instance Windows didukung EBS.

Anda dapat melihat jenis perangkat root dari instance Linux Anda menggunakan salah satu metode berikut.

Console

Untuk menentukan jenis perangkat root dari sebuah instans menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di panel navigasi, pilih Instans, dan pilih instans.
3. Pada tab Penyimpanan, di bawah detail perangkat Root, periksa nilai jenis perangkat Root sebagai berikut:
 - Jika nilainya EBS, ini adalah instans yang didukung Amazon EBS.
 - Jika nilainya INSTANCE-STORE, ini adalah instans yang didukung penyimpanan instans.

AWS CLI

Untuk menentukan jenis perangkat root dari sebuah instans menggunakan baris perintah

Anda dapat menggunakan salah satu dari perintah berikut. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) (AWS Tools for Windows PowerShell)

Ubah volume root agar tetap ada

Secara default, volume root untuk AMI yang didukung oleh Amazon EBS dihapus saat instans berakhir. Anda dapat mengubah perilaku default untuk memastikan bahwa volume tetap ada setelah instans berakhir. Untuk mengubah perilaku default, setel atribut `DeleteOnTermination` menjadi `false` menggunakan pemetaan perangkat blok.

Tugas

- [Mengonfigurasi volume root agar tetap ada selama peluncuran instans](#)
- [Konfigurasi 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 instans menggunakan konsol Amazon EC2 atau alat bantu baris perintah.

Console

Untuk mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instans menggunakan konsol

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 konfigurasi pengaturan jaringan Anda.
4. Untuk Konfigurasi penyimpanan, pilih Lanjutkan.
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 menggunakan AWS CLI

Gunakan perintah [run-instance](#) dan sertakan pemetaan perangkat blok yang menyetel atribut `DeleteOnTermination` ke `false`.

```
aws ec2 run-instances --block-device-mappings file://mapping.json ...other
parameters...
```

Tentukan hal berikut dalam `mapping.json`.

```
[
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

Tools for Windows PowerShell

Untuk mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instance menggunakan Alat untuk Windows PowerShell

Gunakan [New-EC2Instance](#) perintah dan sertakan pemetaan perangkat blok yang menyetel `DeleteOnTermination` atribut ke `false`.

```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.BlockDeviceMapping
C:\> $bdm.DeviceName = "dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> New-EC2Instance -ImageId ami-0abcdef1234567890 -BlockDeviceMapping
$bdm ...other parameters...
```

Konfigurasi volume root agar tetap ada untuk instans yang ada

Anda dapat mengonfigurasi volume root agar tetap ada untuk instans yang berjalan hanya menggunakan alat baris perintah.

AWS CLI

Untuk mengonfigurasi volume root agar tetap ada untuk instance yang ada menggunakan AWS CLI

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
    }
  }
]
```

]

Tools for Windows PowerShell

Konfigurasi volume root agar tetap ada untuk instans yang ada menggunakan AWS Tools for Windows PowerShell

Gunakan [Edit-EC2InstanceAttribute](#) perintah dengan pemetaan perangkat blok yang menyetel `DeleteOnTermination` atribut ke `false`.

```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsInstanceBlockDeviceSpecification
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.InstanceBlockDeviceMappingSpecification
C:\> $bdm.DeviceName = "/dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -BlockDeviceMapping
    $bdm
```

Konfirmasikan bahwa volume root dikonfigurasi agar tetap ada

Anda dapat mengonfirmasi bahwa volume root dikonfigurasi agar tetap ada menggunakan konsol Amazon EC2 atau alat baris perintah.

Console

Anda dapat mengonfirmasi bahwa volume root dikonfigurasi agar tetap ada menggunakan konsol Amazon EC2

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 agar tetap menggunakan AWS CLI

Gunakan perintah [describe-instances](#) dan pastikan bahwa atribut `DeleteOnTermination` di elemen respons `BlockDeviceMappings` diatur ke `false`.

```
aws ec2 describe-instances --instance-id i-1234567890abcdef0
```

```
...
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/sda1",
      "Ebs": {
        "Status": "attached",
        "DeleteOnTermination": false,
        "VolumeId": "vol-1234567890abcdef0",
        "AttachTime": "2013-07-19T02:42:39.000Z"
      }
    }
  ]
...

```

Tools for Windows PowerShell

Untuk mengonfirmasi bahwa volume root dikonfigurasi agar tetap menggunakan AWS Tools for Windows PowerShell

Gunakan [Get-EC2Instance](#) dan verifikasi bahwa DeleteOnTermination atribut dalam elemen BlockDeviceMappings respons diatur ke false.

```
C:\> (Get-EC2Instance -InstanceId i-
i-1234567890abcdef0).Instances.BlockDeviceMappings.Ebs
```

Ubah ukuran awal volume root

Secara default, ukuran volume root ditentukan oleh ukuran snapshot. Anda dapat meningkatkan ukuran awal volume root menggunakan pemetaan perangkat blok dari instans sebagai berikut.

1. Tentukan nama perangkat dari volume root yang ditentukan di AMI, seperti yang dijelaskan di [Lihat volume EBS dalam pemetaan perangkat blok AMI](#).
2. Konfirmasikan ukuran snapshot yang ditentukan dalam pemetaan perangkat blok AMI.
3. Ganti ukuran volume root menggunakan pemetaan perangkat blok instans, seperti yang dijelaskan di [Memperbarui pemetaan perangkat blok saat meluncurkan suatu instans](#), yang menentukan ukuran volume yang lebih besar dari ukuran snapshot.

Sebagai contoh, entri berikut untuk pemetaan perangkat blok instans meningkatkan ukuran volume root, /dev/xvda, hingga 100 GiB. Anda dapat menghilangkan ID snapshot dalam pemetaan perangkat blok instans karena ID snapshot sudah ditentukan dalam pemetaan perangkat blok AMI.

```
{
  "DeviceName": "/dev/xvda",
  "Ebs": {
    "VolumeSize": 100
  }
}
```

Untuk informasi selengkapnya, lihat [Pemetaan perangkat blok](#).

Ganti volume root instans EC2

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 Amazon EBS non-root tetap dilampirkan ke 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.

Topik

- [Bagaimana cara kerjanya?](#)
- [Mengganti volume root](#)
- [Melihat tugas penggantian volume root](#)

Bagaimana cara kerjanya?

Saat Anda mengganti volume root untuk sebuah instans, volume root baru (penggantian) dipulihkan dengan salah satu cara berikut:

- Ke status peluncuran awal — volume dikembalikan ke status awal saat peluncuran instans. Untuk informasi selengkapnya, lihat [Mengembalikan volume root ke status peluncurannya](#).
- Dari snapshot dari garis keturunan yang sama dengan volume root saat ini — hal ini memungkinkan Anda untuk memperbaiki masalah, seperti kerusakan volume root atau kesalahan konfigurasi jaringan sistem operasi tamu. Untuk informasi selengkapnya, lihat [Mengganti volume root menggunakan snapshot](#).
- Dari AMI yang memiliki atribut kunci yang sama dengan instans — hal ini memungkinkan Anda melakukan patching atau upgrade sistem operasi dan aplikasi. Untuk informasi selengkapnya, lihat [Mengganti volume root menggunakan AMI](#).

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. Anda dapat memilih apakah akan menyimpan volume root asli atau tidak 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 menghapusnya secara manual ketika Anda tidak lagi membutuhkannya.

Jika tugas penggantian volume root gagal, instans di-boot ulang dan volume root asli tetap melekat pada instans.

Pertimbangan untuk penggantian volume root

- 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. Semua tipe instans bare metal lainnya tidak didukung.

- Anda dapat menggunakan snapshot apa pun yang termasuk dalam garis keturunan yang sama dengan volume root instans 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.
- Tabel berikut merangkum kemungkinan hasil enkripsi.

	Volume root asli	Snapshot atau AMI 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
Mengembalikan volume root pengganti dari snapshot atau AMI	Dienkripsi	Tidak terenkripsi	Tidak dipertimbangkan	Dienkripsi	Kunci KMS yang sama dengan volume root asli

	Volume root asli	Snapshot atau AMI yang ditentukan	Enkripsi secara default	Volume root pengganti	Kunci enkripsi yang digunakan untuk penggantian volume root
	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 atau AMI 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.

Topik

- [Mengembalikan volume root ke status peluncurannya](#)

- [Mengganti volume root menggunakan snapshot](#)
- [Mengganti volume root menggunakan AMI](#)

Mengembalikan volume root ke status peluncurannya

Anda dapat melakukan penggantian volume root yang menggantikan volume root instans dengan volume root pengganti yang dikembalikan ke status peluncuran volume root asli. Volume pengganti secara otomatis dipulihkan dari snapshot yang digunakan untuk membuat volume asli selama peluncuran instans.

Volume root pengganti mendapatkan tipe, ukuran, dan atribut pengakhiran yang sama dengan volume root asli.

Mengganti volume root menggunakan snapshot

Anda dapat melakukan penggantian volume root yang menggantikan volume root instans dengan volume pengganti yang dikembalikan ke snapshot tertentu. Hal ini memungkinkan Anda untuk memulihkan volume root untuk sebuah instans ke snapshot tertentu yang sebelumnya Anda buat dari volume root tersebut.

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 termasuk dalam garis keturunan yang sama dengan volume root instans saat ini.
- Anda tidak dapat menggunakan salinan snapshot yang dibuat dari snapshot yang diambil dari volume root.
- Setelah berhasil mengganti volume root, snapshot yang diambil dari volume root asli masih dapat digunakan untuk mengganti volume root baru (pengganti).

Mengganti volume root menggunakan AMI

Anda dapat melakukan penggantian volume root menggunakan AMI yang Anda miliki atau AMI yang dibagikan dengan Anda. AMI harus memiliki kode produk, informasi penagihan, tipe arsitektur, dan jenis virtualisasi yang sama seperti yang ada pada instans.

Jika instance diaktifkan untuk ENA atau sriov-net, maka Anda harus menggunakan AMI yang mendukung fitur-fitur tersebut. Jika instans tidak diaktifkan untuk ENA atau sriov-net, maka Anda dapat memilih AMI yang tidak menyertakan dukungan untuk 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 tipe volume yang sama dan atribut hapus saat pengakhiran seperti volume root asli, dan mendapatkan ukuran pemetaan perangkat blok volume root AMI.

Note

Ukuran pemetaan perangkat blok volume root AMI harus sama dengan atau lebih besar dari ukuran volume root asli. Jika ukuran pemetaan perangkat blok volume root AMI lebih kecil dari ukuran volume root asli, permintaan gagal.

Setelah tugas penggantian volume root selesai, informasi baru dan yang diperbarui berikut akan ditampilkan 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 pemetaan perangkat blok volume root AMI harus sama dengan atau lebih besar dari ukuran volume root asli. Jika ukuran pemetaan perangkat blok volume root AMI lebih kecil dari ukuran volume root asli, permintaan gagal.
- Dokumen identitas instans untuk instans diperbarui secara otomatis.
- Jika instans mendukung NitroTPM, data NitroTPM untuk instans direset dan kunci baru dihasilkan.

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. Untuk informasi selengkapnya, lihat [Melihat tugas penggantian volume root](#).

Anda dapat mengganti volume root untuk suatu instans menggunakan salah satu metode berikut.

Note

Jika Anda menggunakan konsol Amazon EC2, fungsionalitas hanya tersedia di konsol baru.

New 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. Di Ganti volume root, lakukan salah satu hal berikut:
 - Untuk mengembalikan volume root pengganti ke status peluncuran awal, pilih Buat tugas pengganti tanpa memilih snapshot.
 - Untuk memulihkan volume root pengganti ke snapshot tertentu, untuk Snapshot, pilih snapshot yang akan digunakan, lalu pilih Buat tugas pengganti.
 - Untuk memulihkan volume root pengganti ke snapshot tertentu menggunakan AMI, untuk AMI, pilih AMI yang akan digunakan, lalu pilih Buat tugas pengganti.
5. Untuk menghapus volume root asli setelah tugas penggantian selesai, pilih Hapus volume root yang diganti.

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`.

```
$ aws ec2 create-replace-root-volume-task \  
--instance-id i-1234567890abcdef0 \  
--delete-replaced-root-volume true
```

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`.

```
$ aws ec2 create-replace-root-volume-task \  
--instance-id i-1234567890abcdef0 \  
--snapshot-id s-1234567890abcdef0 \  
--delete-replaced-root-volume true
```

```
--instance-id i-1234567890abcdef0 \  
--snapshot-id snap-9876543210abcdef0 \  
--delete-replaced-root-volume true
```

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`.

```
$ aws ec2 create-replace-root-volume-task \  
--instance-id i-01234567890abcdef \  
--image-id ami-09876543210abcdef \  
--delete-replaced-root-volume true
```

Tools for Windows 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`.

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -  
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`.

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -  
SnapshotId snap-9876543210abcdef0 -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`.

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -  
ImageId ami-09876543210abcdef -DeleteReplacedRootVolume $true
```

Melihat tugas penggantian volume root

Saat Anda mengganti volume root untuk suatu instans, tugas penggantian volume root dibuat. Tugas penggantian volume root bertransisi melalui status berikut selama proses:

- `pending` — volume pengganti sedang dibuat.
- `in-progress` — volume asli sedang terlepas dan volume penggantian sedang dilampirkan.
- `succeeded` — volume pengganti telah berhasil dilampirkan ke instans dan instans tersedia.
- `failing` — tugas penggantian adalah dalam proses gagal.
- `failed` — tugas penggantian telah gagal tetapi volume root asli masih terpasang.
- `failing-detached` — tugas penggantian ada dalam proses gagal dan instans mungkin tidak melampirkan volume root.
- `failed-detached` — tugas pengganti telah gagal dan instans tidak memiliki volume root yang terpasang.

Anda dapat melihat tugas penggantian volume root untuk suatu instans menggunakan salah satu metode berikut.

Note

Jika Anda menggunakan konsol Amazon EC2, fungsionalitas hanya tersedia di konsol baru.

Console

Untuk melihat tugas penggantian volume root

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Di panel navigasi, pilih Instans.
3. Pilih instans untuk yang digunakan untuk melihat tugas penggantian volume root, kemudian pilih tab Penyimpanan.
4. Di tab Penyimpanan, perluas Tugas penggantian volume root terakhir.

AWS CLI

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
```

```
{  
  "ReplaceRootVolumeTasks": [  
    {  
      "ReplaceRootVolumeTaskId": "replacevol-1234567890abcdef0",  
      "InstanceId": "i-1234567890abcdef0",  
      "TaskState": "succeeded",  
      "StartTime": "2020-11-06 13:09:54.0",  
      "CompleteTime": "2020-11-06 13:10:14.0",  
      "SnapshotId": "snap-01234567890abcdef",  
      "DeleteReplacedRootVolume": "True"  
    }  
  ]  
}
```

Atau, tentukan filter `instance-id` untuk menyaring hasil menurut instans.

```
$ aws ec2 describe-replace-root-volume-tasks \  
--filters Name=instance-id,Values=i-1234567890abcdef0
```

Tools for Windows PowerShell

Untuk melihat status tugas penggantian volume root

Gunakan [Get-EC2ReplaceRootVolumeTask](#) perintah dan tentukan ID tugas penggantian volume root untuk dilihat.

```
PS C:\> Get-EC2ReplaceRootVolumeTask -  
ReplaceRootVolumeTaskIds replacevol-1234567890abcdef0
```

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 di instans Amazon EC2

Saat Anda memasang volume ke instans, Anda menyertakan nama perangkat untuk volume tersebut. Nama perangkat ini digunakan oleh Amazon EC2 Driver perangkat blok untuk instans tersebut menetapkan nama volume aktual saat memasang volume, dan nama yang ditetapkan dapat berbeda dari nama yang digunakan oleh Amazon EC2.

Jumlah volume yang dapat didukung oleh instans Anda mendukung ditentukan oleh sistem operasi. Untuk informasi selengkapnya, lihat [Batasan volume instans](#).

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 tipe instans mendukung AMIs HVM. Beberapa tipe instans generasi sebelumnya mendukung PV AMI. 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 AMI](#).

Tabel berikut mencantumkan nama perangkat yang tersedia yang dapat Anda tentukan dalam pemetaan perangkat blok atau saat melampirkan volume EBS.

Tipe virtualisasi	Tersedia	Terpesan untuk volume root	Direkomen dasikan untuk volume EBS	Volume penyimpanan instans
Paravirtual	/dev/sd[a-z] /dev/sd[a-z] [1-15] /dev/hh[a-z] /dev/hh[a-z] [1-15]	/dev/sda1	/dev/sd[f-p] /dev/sd[f-p][1-6]	/dev/sd[b-e]
HVM	/dev/sd[a-z] /dev/xvd [a-d] [a-z] /dev/xvd [e-z]	Dibedakan berdasarkan AMI /dev/sda1 atau /dev/xvda	/dev/sd[f-p]	/dev/sd[b-e] /dev/sd[b-h] (h1.16xlarge) /dev/sd[b-y] (d2.8xlarge) /dev/sd[b-i] (i2.8xlarge) **

* Nama perangkat yang Anda tentukan untuk volume NVMe EBS dalam pemetaan perangkat blok diganti namanya menggunakan nama perangkat NVMe (/dev/nvme [0-26]n1). Driver perangkat blok dapat menentukan nama perangkat NVMe dengan urutan yang berbeda dari yang Anda tentukan untuk volume dalam pemetaan perangkat blok.

** Volume penyimpanan instans NVMe secara otomatis dienumerasi dan diberi nama perangkat NVMe.

Instans Windows

AMI Windows menggunakan salah satu set driver berikut untuk mengizinkan akses ke perangkat keras virtual: AWS PV, Citrix PV, dan PV. RedHat Untuk informasi selengkapnya, lihat [the section called “Driver Windows PV”](#).

Tabel berikut mencantumkan nama perangkat yang tersedia yang dapat Anda tentukan dalam pemetaan perangkat blok atau saat melampirkan volume EBS.

Jenis driver	Tersedia	Terpesan untuk volume root	Direkomen dasikan untuk volume EBS	Volume penyimpanan instans
AWS PV, Citrix PV	xvd[b-z]	/dev/sda1	xvd[f-z] *	xvdc[a-x]
	xvd[b-c][a-z]			xvd[a-e]
	/dev/sda1			**
	/dev/sd[b-e]			
Red Hat PV	xvd[a-e]	/dev/sda1	xvd[f-z] *	xvdc[a-x]
	xvd[b-c][a-z]			xvd[a-e]
	/dev/sda1			
	/dev/sd[b-e]			

* Untuk Citrix PV dan Red Hat PV, jika Anda memetakan volume EBS dengan nama xvda, Windows tidak mengenali volume (volume terlihat untuk AWS PV atau AWS NVMe).

** Volume penyimpanan instans NVMe dinumerasi secara otomatis dan diberi huruf drive Windows.

Untuk informasi selengkapnya tentang volume penyimpanan instans, lihat [Penyimpanan instans Amazon EC2](#). Untuk informasi selengkapnya tentang volume NVMe EBS (instans berbasis Nitro), termasuk cara mengidentifikasi perangkat EBS, lihat [Amazon EBS dan NVMe di Panduan Pengguna Amazon EBS](#).

Pertimbangan nama perangkat

Ingatlah hal-hal berikut ini saat memilih nama perangkat:

- 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 instans NVMe untuk suatu instans bergantung pada ukuran instans. Volume penyimpanan instans NVMe secara otomatis disebutkan dan diberi nama perangkat NVMe (instance Linux) atau huruf drive Windows (instance Windows).
- (Windows instance) AWS Windows AMI hadir dengan perangkat lunak tambahan yang menyiapkan instance saat pertama kali boot. Ini adalah layanan EC2Config (Windows AMI sebelum Windows Server 2016) atau EC2Launch (Windows Server 2016 dan versi yang lebih tinggi). 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 [Konfigurasi pengaturan peluncuran untuk instans Windows Amazon EC2](#), dan [Petakan disk ke volume pada instans Windows](#).
- (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.
- (Instans Linux) AMI HVM tidak mendukung penggunaan nomor tambahan pada nama perangkat, kecuali untuk /dev/sda1, yang dicadangkan untuk perangkat root, dan /dev/sda2. Meskipun /dev/sda2 dapat digunakan, kami tidak menyarankan penggunaan pemetaan perangkat ini dengan instans HVM.
- (Instance Linux) Saat menggunakan AMI PV, 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 dipilih, 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).

Pemetaan perangkat blok

Setiap instans yang Anda luncurkan memiliki volume perangkat root yang terkait, yang bisa berupa 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 instans saat diluncurkan. Anda juga dapat melampirkan volume EBS tambahan ke instance yang sedang berjalan. Namun, satu-satunya cara untuk melampirkan volume penyimpanan instans ke instans adalah dengan menggunakan pemetaan perangkat blok untuk melampirkan volume saat instans diluncurkan.

Daftar Isi

- [Konsep pemetaan perangkat blok](#)
- [Pemetaan perangkat blok AMI](#)
- [Pemetaan perangkat blok instans](#)

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 I/O buffer. Contohnya termasuk hard disk, drive CD-ROM, 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 di instans Amazon EC2](#).

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-xxxxxxx). 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 perangkat root dan `false` untuk volume yang 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 operasi input/output per detik (IOPS) yang didukung oleh volume. (Hanya digunakan dengan volume `io1` dan `io2`.)

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.

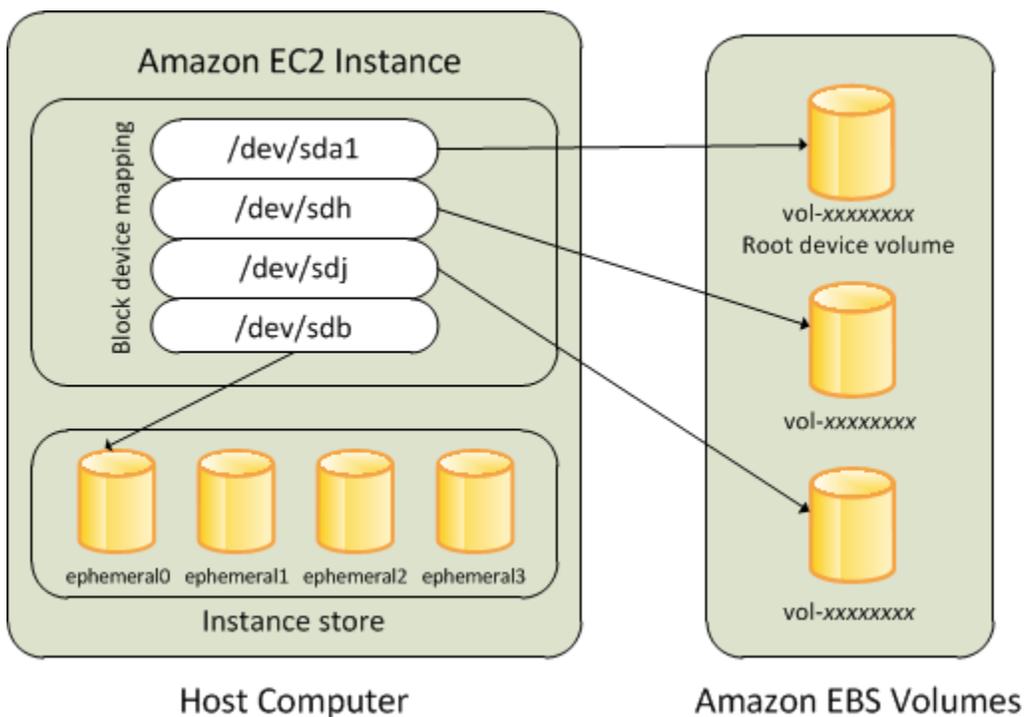
Note

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 instans yang didukung EBS. Gambar ini memetakan `/dev/sdb` ke `ephemeral0` dan memetakan dua volume EBS, satu untuk `/dev/sdh` dan yang lainnya ke `/dev/sdj`. Gambar ini juga menunjukkan volume EBS yang merupakan volume perangkat root, `/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.

Instance Linux — Nama perangkat yang ditentukan dalam pemetaan perangkat blok dipetakan ke perangkat blok yang sesuai saat instance pertama kali melakukan booting. 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 instans Amazon 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 saat instance pertama kali melakukan boot, dan kemudian layanan Ec2Config menginisialisasi dan memasang drive. Volume perangkat 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, lihat [the section called “Konfigurasi agen peluncuran Windows”](#).

Pemetaan perangkat blok 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 AMI yang didukung EBS, Anda dapat menambahkan volume EBS dan volume penyimpanan instans menggunakan pemetaan perangkat blok. Untuk AMI yang didukung penyimpanan instans, Anda dapat menambahkan volume penyimpanan instans hanya dengan memodifikasi entri pemetaan perangkat blok di file manifes image 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 menggunakan konsol

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.

Command line

Untuk menambahkan volume ke AMI menggunakan baris perintah

Gunakan AWS CLI perintah [create-image](#) untuk menentukan pemetaan perangkat blok untuk AMI yang didukung EBS. Gunakan AWS CLI perintah [register-image](#) untuk menentukan pemetaan perangkat blok untuk instance store-backed AMI.

Tentukan pemetaan perangkat blok menggunakan parameter `--block-device-mappings`. Argumen yang dikodekan dalam JSON dapat diberikan secara langsung pada baris perintah atau dengan referensi ke file:

```
--block-device-mappings [mapping, ...]  
--block-device-mappings [file://mapping.json]
```

Untuk menambahkan volume penyimpanan instans, gunakan pemetaan berikut.

```
{  
  "DeviceName": "device_name",  
  "VirtualName": "ephemeral0"  
}
```

Untuk menambahkan volume gp2 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-xxxxxxxx"  
  }  
}
```

Untuk pemetaan perangkat, gunakan pemetaan berikut.

```
{
  "DeviceName": "device_name",
  "NoDevice": ""
}
```

Atau, Anda dapat menggunakan parameter `-BlockDeviceMapping` dengan perintah berikut (AWS Tools for Windows PowerShell):

- [New-EC2Image](#)
- [Register-EC2Image](#)

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 menggunakan konsol

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih AMI.
3. Pilih citra EBS dari daftar Filter untuk mendapatkan daftar AMI yang didukung oleh EBS.
4. Pilih AMI yang diinginkan, dan lihat tab Detail. Minimal, informasi berikut ini tersedia untuk perangkat 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.

Command line

Untuk melihat volume EBS untuk AMI menggunakan baris perintah

Gunakan perintah [describe-images](#) (AWS CLI) atau perintah [Get-EC2Image](#)(AWS Tools for Windows PowerShell) untuk menghitung volume EBS dalam pemetaan perangkat blok untuk AMI.

Pemetaan perangkat blok instans

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 menempa 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.

Daftar Isi

- [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 menambahkan volume ke suatu instans menggunakan konsol

1. Buka konsol Amazon EC2.

2. Dari dasbor, pilih Luncurkan instans.
3. Di halaman Pilih Amazon Machine Image (AMI), pilih AMI yang akan digunakan dan pilih Pilih.
4. Ikuti wizard untuk menyelesaikan halaman Pilih Tipe Instans dan Konfigurasi Detail Instans.
5. Di halaman Tambahkan Penyimpanan Anda dapat memodifikasi volume root, volume EBS, dan volume penyimpanan instans sebagai berikut:
 - Untuk mengubah ukuran volume root, temukan volume Root dalam kolom Tipe, dan ubah bidang Ukuran.
 - Untuk menekan volume EBS yang ditentukan oleh pemetaan perangkat blok AMI yang digunakan untuk meluncurkan instans, temukan volume, dan klik ikon Hapus.
 - Untuk menambahkan volume EBS, pilih Tambahkan Volume Baru, pilih EBS dari daftar Tipe, dan isi bidang (Perangkat, Snapshot, dan sebagainya).
 - Untuk menekan penyimpanan instans yang ditentukan oleh pemetaan perangkat blok AMI yang digunakan untuk meluncurkan instans, temukan volum, dan pilih ikon Hapus.
 - Untuk menambahkan volume penyimpanan instan, pilih Tambahkan Volume Baru, pilih Penyimpanan Instans dari daftar Tipe dan pilih nama perangkat dari Perangkat.
6. Selesaikan halaman wizard yang tersisa, dan pilih Luncurkan.

Command line

Untuk menambahkan volume ke instance menggunakan AWS CLI

Gunakan AWS CLI perintah [run-instance](#) dengan `--block-device-mappings` opsi untuk menentukan pemetaan perangkat blok untuk instance saat peluncuran.

Misalnya, AMI yang didukung EBS menentukan pemetaan perangkat blok berikut untuk instance Linux:

- `/dev/sdb = ephemeral0`
- `/dev/sdh = snap-1234567890abcdef0`
- `/dev/sdj = 100`

Untuk mencegah `/dev/sdj` terpasang pada instans yang diluncurkan dari AMI ini, gunakan pemetaan berikut.

```
{
  "DeviceName": "/dev/sdj",
  "NoDevice": ""
}
```

Untuk meningkatkan ukuran `/dev/sdh` to 300 GiB, tentukan pemetaan berikut. Perhatikan bahwa Anda tidak perlu menentukan ID snapshot untuk `/dev/sdh`, karena menentukan nama perangkat sudah cukup untuk mengidentifikasi volume.

```
{
  "DeviceName": "/dev/sdh",
  "Ebs": {
    "VolumeSize": 300
  }
}
```

Untuk meningkatkan ukuran volume root saat peluncuran instans, panggil [describe-image](#) terlebih dahulu dengan ID AMI untuk memverifikasi nama perangkat volume root. Sebagai contoh, `"RootDeviceName": "/dev/xvda"`. Untuk mengganti ukuran volume root, tentukan nama perangkat dari perangkat root yang digunakan oleh AMI dan ukuran volume yang baru.

```
{
  "DeviceName": "/dev/xvda",
  "Ebs": {
    "VolumeSize": 100
  }
}
```

Untuk memasang volume penyimpanan instans tambahan, `/dev/sdc`, tetapkan pemetaan berikut ini. Jika tipe instans tidak mendukung volume penyimpanan instans banyak instans, pemetaan ini tidak berpengaruh. Jika instans mendukung volume penyimpanan instans NVMe, maka volume tersebut akan secara otomatis dicacah dan diberi nama perangkat NVMe.

```
{
  "DeviceName": "/dev/sdc",
  "VirtualName": "ephemeral1"
}
```

Untuk menambahkan volume ke instance menggunakan AWS Tools for Windows PowerShell

Gunakan `-BlockDeviceMapping` parameter dengan [New-EC2Instance](#) perintah (AWS Tools for Windows PowerShell).

Perbarui pemetaan perangkat blok instans yang berjalan

Anda dapat menggunakan AWS CLI perintah [modify-instance-attribute](#) untuk memperbarui pemetaan perangkat blok dari instance yang sedang berjalan. Anda tidak perlu menghentikan instans sebelum mengubah atribut ini.

```
aws ec2 modify-instance-attribute --instance-id i-1a2b3c4d --block-device-mappings
file://mapping.json
```

Misalnya, untuk menjaga volume root pada saat pengakhiran instans, tentukan hal berikut di `mapping.json`.

```
[
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

Atau, Anda dapat menggunakan `-BlockDeviceMapping` parameter dengan [Edit-EC2InstanceAttribute](#) perintah (AWS Tools for Windows PowerShell).

Lihat volume EBS dalam pemetaan perangkat blok instans

Anda dapat dengan mudah mengenumerasi volume EBS yang dipetakan sebagai instans.

Note

Untuk instance yang diluncurkan sebelum rilis API 2009-10-31, tidak AWS dapat menampilkan pemetaan perangkat blok. Anda harus melepaskan dan memasang kembali volume sehingga AWS dapat menampilkan pemetaan perangkat blok.

Console

Untuk melihat volume EBS untuk instans menggunakan konsol

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih Contoh.
3. Dalam kotak pencarian, masukkan Tipe perangkat root, lalu pilih EBS. Ini menampilkan daftar instans yang didukung EBS.
4. Pilih instans yang diinginkan dan lihat detail yang ditampilkan di tab Penyimpanan. Minimal, informasi berikut ini tersedia untuk perangkat 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.

5. Untuk menampilkan informasi tambahan tentang volume EBS, pilih ID volumenya untuk membuka halaman volume.

Command line

Untuk melihat volume EBS untuk instans menggunakan baris perintah

Gunakan perintah [describe-instances](#) (AWS CLI) atau perintah [Get-EC2Instance](#) (AWS Tools for Windows PowerShell) untuk menghitung volume EBS dalam pemetaan perangkat blok untuk sebuah instance.

Lihat pemetaan perangkat blok instans untuk volume penyimpanan instans

Tipe instans menentukan jumlah dan jenis volume penyimpanan instance 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 instance untuk instans Anda, jalankan lsblk perintah (instance Linux) atau buka Windows Disk Management (instance 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 Namespace	SN	Model	
----- -----			
/dev/nvme0n1	vol106afc3f8715b7a597	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 mencantumkan volume NVMe EBS dan penyimpanan instance. Untuk informasi selengkapnya, lihat [the section called “Mencantumkan volume NVMe”](#).

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 [Cara menggunakan metadata instans](#).

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
```

Perangkat `ami` adalah perangkat root seperti yang terlihat oleh instans. 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
```

Perangkat `ami` adalah perangkat `root` seperti yang terlihat oleh instans. 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
```

Petakan disk ke volume pada instans Windows

Note

Topik ini hanya berlaku untuk instance Windows.

Instans Windows Anda dilengkapi dengan volume EBS yang berfungsi sebagai volume root. Jika instans Windows Anda menggunakan driver AWS PV atau Citrix PV, Anda dapat menambahkan hingga 25 volume secara opsional, membuat total 26 volume. Untuk informasi selengkapnya, lihat [Batasan volume instans](#).

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, lihat [Nama perangkat di instans Amazon EC2](#). 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. Jika Anda meluncurkan instance dari AMI Windows yang menggunakan driver AWS PV atau Citrix PV, Anda dapat menggunakan hubungan yang dijelaskan di halaman ini untuk memetakan disk Windows Anda ke penyimpanan instans dan volume EBS Anda. Jika Windows AMI Anda menggunakan driver PV Red Hat, Anda dapat memperbarui instans Anda untuk menggunakan driver Citrix. Untuk informasi selengkapnya, lihat [the section called "Mutakhirkan driver PV"](#).

Daftar Isi

- [Mencantumkan volume NVMe](#)
 - [Mencantumkan disk NVMe menggunakan Manajemen Disk](#)
 - [Daftar disk NVMe menggunakan PowerShell](#)
 - [Memetakan volume NVMe EBS](#)
- [Mencantumkan volume](#)
 - [Mencantumkan disk menggunakan Manajemen Disk](#)
 - [Memetakan perangkat disk ke nama perangkat](#)

- [Volume penyimpanan instans](#)
- [Volume EBS](#)
- [Daftar disk menggunakan PowerShell](#)

Mencantumkan volume NVMe

Anda dapat menemukan disk di instans Windows menggunakan Manajemen Disk atau Powershell.

Mencantumkan disk NVMe menggunakan Manajemen Disk

Anda dapat menemukan disk di instans Windows Anda menggunakan Manajemen Disk.

Untuk menemukan disk di instans Windows Anda

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Windows Anda](#).
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.

Disk Management [-] [□] [×]

File Action View Help

← → [📅] [?] [📄] [🗨] [✓] [📧]

Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% Free
(C:)	Simple	Basic	NTFS	Healthy (S...	30.00 GB	13.22 GB	44 %
New Volume (D:)	Simple	Basic	NTFS	Healthy (P...	8.00 GB	7.97 GB	100 %
New Volume (E:)	Simple	Basic	NTFS	Healthy (P...	8.00 GB	7.97 GB	100 %
New Volume (F:)	Simple	Basic	NTFS	Healthy (P...	279.39 GB	279.28 GB	100 %
New Volume (G:)	Simple	Basic	NTFS	Healthy (P...	279.39 GB	279.28 GB	100 %

Disk 0 Basic 30.00 GB Online	(C:) 30.00 GB NTFS Healthy (System, Boot, Page File, Active, Crash Dump, Primary Partition)
Disk 1 Basic 8.00 GB Online	New Volume (D:) 8.00 GB NTFS Healthy (Primary Partition)
Disk 2 Basic 8.00 GB Online	New Volume (E:) 8.00 GB NTFS Healthy (Primary Partition)
Disk 3 Basic 279.40 GB Online	New Volume (F:) 279.39 GB NTFS Healthy (Primary Partition)
Disk 4 Basic 279.40 GB Online	New Volume (G:) 279.39 GB NTFS Healthy (Primary Partition)

Unallocated
 Primary partition

Daftar disk NVMe menggunakan PowerShell

PowerShell Skrip berikut mencantumkan setiap disk dan nama dan volume perangkat yang sesuai. Ini dimaksudkan untuk digunakan dengan [instance yang dibangun di atas Sistem AWS Nitro](#), yang menggunakan NVMe EBS dan volume penyimpanan instance.

Connect ke instance Windows Anda dan jalankan perintah berikut untuk mengaktifkan eksekusi PowerShell script.

```
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-6AAD8C2AE1193F0	Ephemeral	N/A	Temporary
Storage						
4	1	G	AWS-13E7299C2BD031A28	Ephemeral	N/A	Temporary
Storage						

Jika Anda tidak mengonfigurasi kredensi Anda untuk Alat untuk Windows PowerShell pada instance Windows, skrip tidak dapat memperoleh ID volume EBS dan menggunakan N/A di kolom. EbsVolumeId

Memetakan volume NVMe EBS

Dengan [instans yang dibangun di Sistem AWS Nitro](#), volume EBS diekspos sebagai perangkat NVMe. Anda dapat menggunakan perintah [Get-Disk](#) untuk memetakan nomor disk Windows ke ID volume EBS.

```
PS C:\> Get-Disk
Number Friendly Name Serial Number HealthStatus
OperationalStatus Total Size Partition
Style
-----
-----
-----
3 NVMe Amazo... AWS6AAD8C2AE1193F0_00000001. Healthy Online
279.4 GB MBR
```

4	NVMe Amazo... AWS13E7299C2BD031A28_00000001. 279.4 GB MBR	Healthy	Online
2	NVMe Amazo... vol0a4064b39e5f534a2_00000001. 8 GB MBR	Healthy	Online
0	NVMe Amazo... vol03683f1d861744bc7_00000001. 30 GB MBR	Healthy	Online
1	NVMe Amazo... vol082b07051043174b9_00000001. 8 GB MBR	Healthy	Online

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
```

Mencantumkan volume

Anda dapat menemukan disk di instans Windows menggunakan Manajemen Disk atau Powershell.

Mencantumkan disk menggunakan Manajemen Disk

Anda dapat menemukan disk di instans Windows Anda menggunakan Manajemen Disk.

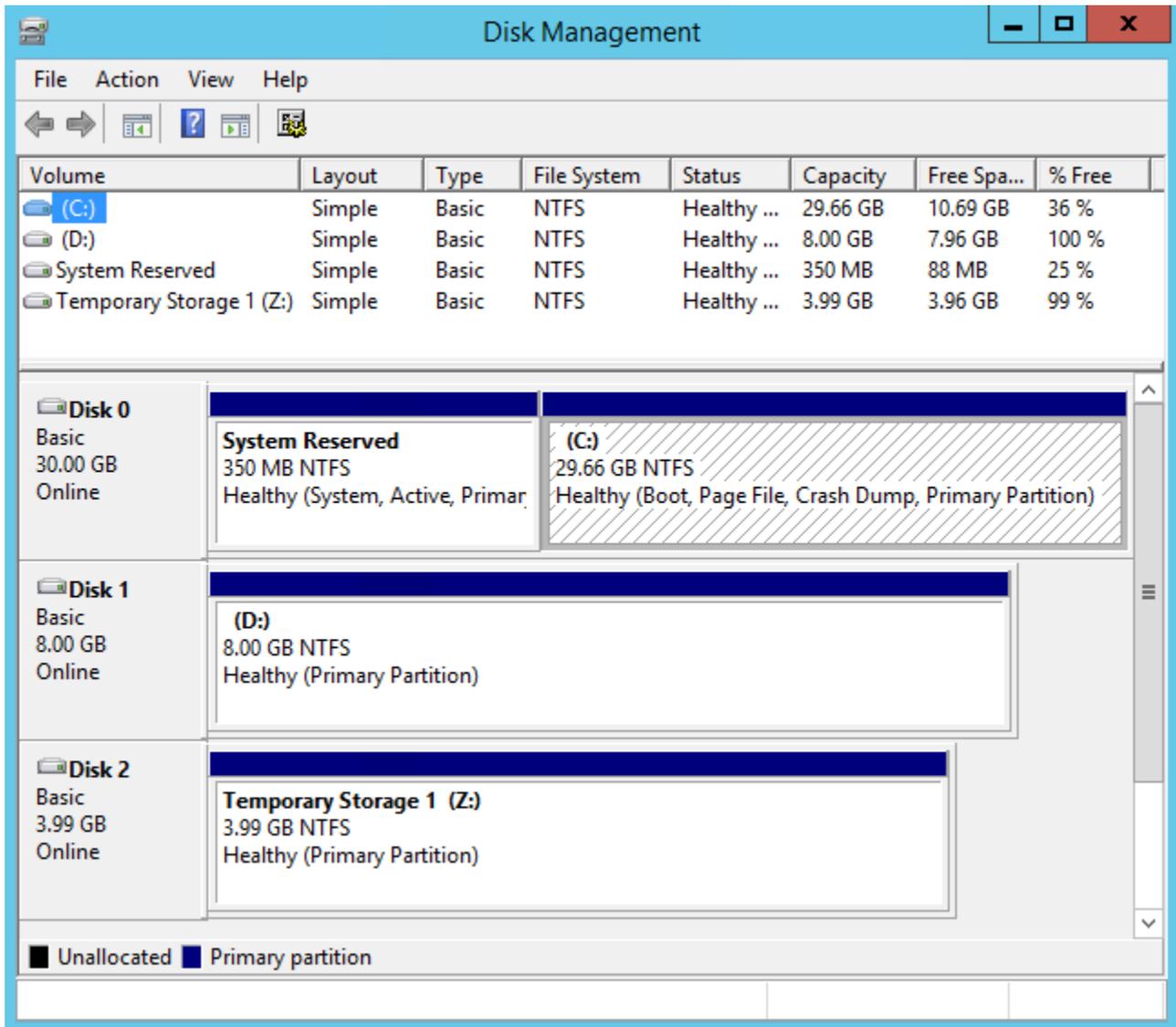
Untuk menemukan disk di instans Windows Anda

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Windows Anda](#).
2. Mulai utilitas Manajemen Disk.

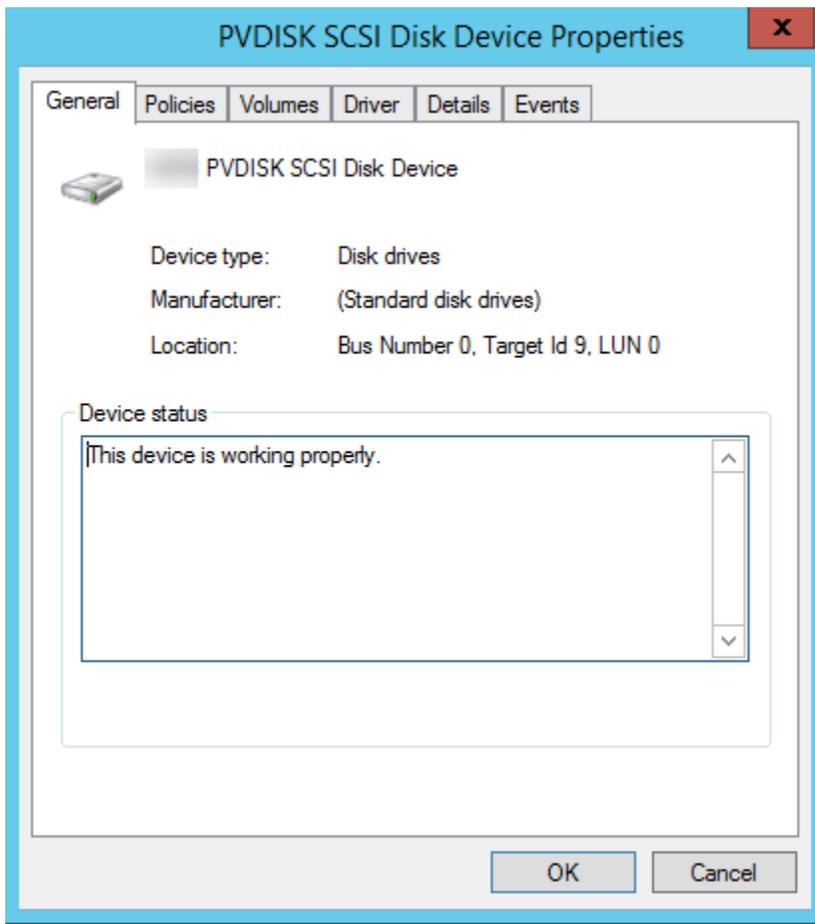
Pada bilah tugas, klik kanan logo Windows, lalu pilih Manajemen Disk.

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. Klik kanan panel abu-abu dengan label Disk 1, lalu pilih Properti. Perhatikan nilai Lokasi dan cari dalam tabel di [Memetakan perangkat disk ke nama perangkat](#). 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.



Memetakan perangkat disk ke nama perangkat

Driver perangkat blok untuk instans menetapkan nama volume aktual saat melakukan pemasangan volume.

Pemetaan

- [Volume penyimpanan instans](#)
- [Volume EBS](#)

Volume penyimpanan instans

Tabel berikut menjelaskan bagaimana driver Citrix PV dan PV memetakan AWS volume penyimpanan instans non-NVMe ke volume Windows. Jumlah volume penyimpanan instans yang tersedia ditentukan oleh tipe instans. Untuk informasi selengkapnya, lihat [Volume penyimpanan instans](#).

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

Volume EBS

Tabel berikut menjelaskan bagaimana driver Citrix PV dan PV memetakan AWS 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

Lokasi	Nama perangkat
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
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

Lokasi	Nama perangkat
Nomor Bus 0, ID Target 24, LUN 0	xkertas
Nomor Bus 0, ID Target 25, LUN 0	xvdz

Daftar disk menggunakan 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 Alat untuk PowerShell, atau melampirkan peran IAM ke instance.
- Tidak mendukung volume NVMe.
- Tidak mendukung disk dinamis.

Connect ke instance Windows Anda dan jalankan perintah berikut untuk mengaktifkan eksekusi PowerShell script.

```
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 Anda pada instans Windows, skrip tidak bisa mendapatkan ID volume EBS dan menggunakan N/A pada kolom EbsVolumeId.

Snapshot Amazon EBS berbasis Windows VSS yang konsisten dengan aplikasi

Note

Snapshot berbasis Windows VSS yang konsisten aplikasi hanya didukung dengan instance Windows.

[Anda dapat mengambil snapshot yang konsisten dengan aplikasi dari semua volume Amazon EBS yang dilampirkan ke instans Windows Amazon EC2 Anda menggunakan Run Command.AWS Systems Manager](#) Proses snapshot menggunakan [Layanan Salinan Snapshot \(VSS\) Volume](#) Windows untuk melakukan pencadangan tingkat volume EBS pada aplikasi sadar VSS. 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 cara menagih snapshot Amazon EBS EBS saya?](#)

Daftar Isi

- [Apa itu VSS?](#)
- [Prasyarat](#)
- [Buat snapshot EBS yang didukung VSS](#)
- [Memecahkan masalah snapshot EBS berbasis Windows VSS](#)
- [Pulihkan volume EBS dari snapshot EBS yang mendukung VSS](#)
- [AWS riwayat versi VSS solution](#)

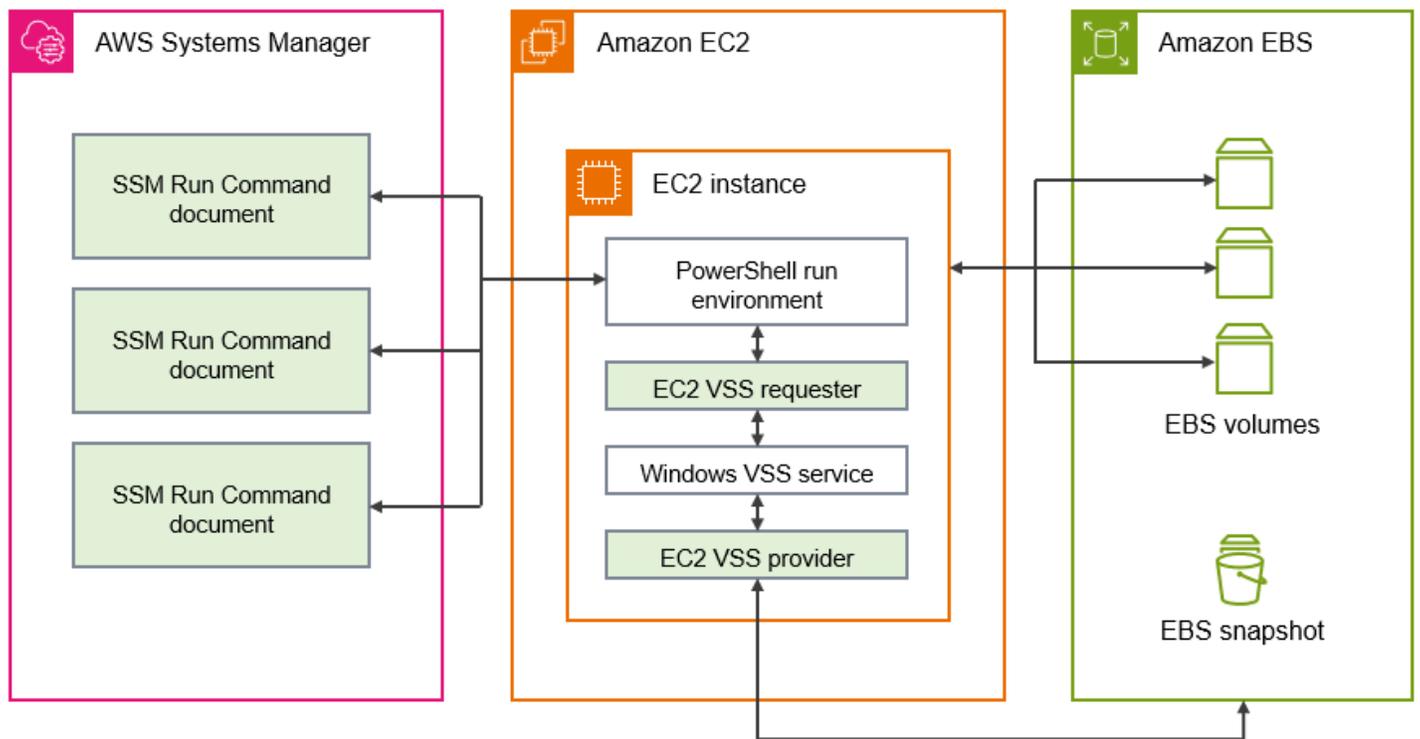
Apa itu VSS?

Layanan Salinan Snapshot (VSS) Volume adalah teknologi pencadangan 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 Systems Manager (SSM) yang memfasilitasi pembuatan cadangan, dan [paket Distributor Systems Manager](#), yang disebut *AwsVssComponents*, yang mencakup pemohon VSS EC2 dan penyedia VSS EC2. 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 untuk mengambil skrip snapshot EBS berbasis VSS yang konsisten aplikasi terdiri dari langkah-langkah berikut.

1. Selesaikan [Prasyarat](#).
2. Masukkan parameter untuk dokumen SSM `AWSEC2-VssInstallAndSnapshot` dan jalankan dokumen ini menggunakan Run Command. Untuk informasi selengkapnya, lihat [Jalankan dokumen `VssInstallAndSnapshot` perintah `AWSEC 2` \(disarankan\)](#).
3. Layanan VSS Windows pada instans Anda mengoordinasikan semua operasi I/O yang sedang berjalan untuk menjalankan aplikasi.
4. Sistem akan membersihkan semua buffer I/O dan menjeda sementara 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 dicabut dan I/O melanjutkan operasi.
7. Sistem menambahkan semua snapshot yang baru dibuat ke daftar snapshot EBS. Sistem menandai semua snapshot EBS berkemampuan VSS yang berhasil dibuat oleh proses ini dengan: `true`. `AppConsistent`

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 [Pulihkan volume EBS dari snapshot EBS yang mendukung VSS](#).

Prasyarat

Anda dapat membuat snapshot EBS berbasis VSS dengan Systems Manager Run Command AWS Backup, atau Amazon Data Lifecycle Manager. Prasyarat berikut berlaku untuk semua solusi.

Prasyarat

- [Persyaratan sistem](#)
- [Izin IAM](#)
- [Komponen VSS](#)

Persyaratan sistem

Instal Agen Systems Manager

VSS diatur oleh (Systems AWS Systems Manager Manager) 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 instance yang menjalankan Windows Server 2012 dan yang lebih baru. Untuk versi Windows yang lebih lama, lihat tabel dukungan versi Windows di [AWS riwayat versi VSS solution](#).

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 instans Anda menggunakan versi sebelumnya dari .NET Framework, Anda harus menginstal versi 4.6 atau yang lebih baru menggunakan Windows Update.

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 AWS Tools for Windows PowerShell Pengguna.

PowerShell Versi Windows

Pastikan instans Anda menjalankan Windows versi PowerShell mayor3,4, atau5. Untuk memeriksa versi Anda, jalankan perintah berikut di PowerShell terminal pada instance.

```
C:\> $PSVersionTable.PSVersion
```

PowerShell modus bahasa

Pastikan instans Anda memiliki mode PowerShell bahasa yang disetel keFullLanguage. Untuk informasi selengkapnya, lihat [about_Language_Modes](#) di dokumentasi Microsoft.

Izin IAM

Peran IAM yang dilampirkan ke instans Amazon EC2 Windows Anda harus memiliki izin untuk membuat snapshot yang konsisten dengan aplikasi dengan VSS. Untuk memberikan izin yang diperlukan, Anda dapat melampirkan AWSEC2VssSnapshotPolicy kebijakan ke profil instans Anda.

Kebijakan ini memungkinkan Systems Manager untuk melakukan tindakan berikut:

- Buat dan beri tag snapshot EBS
- Membuat dan menandai Amazon Machine Images (AMI)
- Lampirkan metadata, seperti ID perangkat, ke tag snapshot default yang dibuat VSS.

Topik

- [Lampirkan kebijakan snapshot berkemampuan VSS ke profil instans Anda](#)
- [Kebijakan terkelola untuk membuat snapshot VSS](#)
- [Kebijakan lama \(tidak lagi didukung\)](#)

Lampirkan kebijakan snapshot berkemampuan VSS ke profil instans Anda

Untuk memberikan izin untuk snapshot berkemampuan VSS untuk instans Anda, Anda melampirkan kebijakan AWSEC2VssSnapshotPolicy terkelola ke peran profil instans Anda sebagai berikut. Penting untuk memastikan bahwa instans Anda memenuhi semua [Persyaratan sistem](#).

 Note

Untuk menggunakan kebijakan terkelola, instans Anda harus memiliki versi `AwsVssComponents` paket 2.3.1 atau yang lebih baru diinstal. Untuk riwayat versi, lihat [AwsVssComponents versi paket](#).

Jika Anda memiliki versi `AwsVssComponents` paket sebelumnya yang diinstal pada instance Anda, lihat [Kebijakan warisan](#).

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.

Kebijakan terkelola untuk membuat snapshot VSS

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 `AWSEC2VssSnapshotPolicy` kebijakan, kebijakan terkelola, 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 (AMI) dan Snapshots EBS. Untuk melampirkan kebijakan, lihat [Lampirkan kebijakan snapshot berkemampuan VSS ke profil instans Anda](#).

Izin diberikan oleh AWSEC2VssSnapshotPolicy

AWSEC2VssSnapshotPolicyKebijakan ini mencakup izin Amazon EC2 berikut:

- ec2: CreateTags — Tambahkan tag ke snapshot EBS dan AMI untuk membantu mengidentifikasi dan mengkategorikan sumber daya.
- ec2: DescribeInstance Atribut — 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 AMI dan snapshot EC2.
- ec2: DescribeSnapshots — Tentukan waktu pembuatan dan status snapshot untuk memverifikasi konsistensi aplikasi.

Contoh kebijakan

Berikut ini adalah contoh AWSEC2VssSnapshotPolicy kebijakan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DescribeInstanceInfo",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstanceAttribute"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:instance/*"
      ],
      "Condition": {
        "StringLike": {
          "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
        }
      }
    }
  ],
}
```

```
{
  "Sid": "CreateSnapshotsWithTag",
  "Effect": "Allow",
  "Action": [
    "ec2:CreateSnapshots"
  ],
  "Resource": [
    "arn:aws:ec2:*:*:snapshot/*"
  ],
  "Condition": {
    "StringLike": {
      "aws:RequestTag/AwsVssConfig": "*"
    }
  }
},
{
  "Sid": "CreateSnapshotsAccessInstance",
  "Effect": "Allow",
  "Action": [
    "ec2:CreateSnapshots"
  ],
  "Resource": [
    "arn:aws:ec2:*:*:instance/*"
  ],
  "Condition": {
    "StringLike": {
      "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
    }
  }
},
{
  "Sid": "CreateSnapshotsAccessVolume",
  "Effect": "Allow",
  "Action": [
    "ec2:CreateSnapshots"
  ],
  "Resource": [
    "arn:aws:ec2:*:*:volume/*"
  ]
},
{
  "Sid": "CreateImageWithTag",
  "Effect": "Allow",
  "Action": [
```

```

        "ec2:CreateImage"
    ],
    "Resource": [
        "arn:aws:ec2:*:*:snapshot/*",
        "arn:aws:ec2:*:*:image/*"
    ],
    "Condition": {
        "StringLike": {
            "aws:RequestTag/AwsVssConfig": "*"
        }
    }
},
{
    "Sid": "CreateImageAccessInstance",
    "Effect": "Allow",
    "Action": [
        "ec2:CreateImage"
    ],
    "Resource": [
        "arn:aws:ec2:*:*:instance/*"
    ],
    "Condition": {
        "StringLike": {
            "ec2:SourceInstanceARN": "*${ec2:InstanceId}"
        }
    }
},
{
    "Sid": "CreateTagsOnResourceCreation",
    "Effect": "Allow",
    "Action": "ec2:CreateTags",
    "Resource": [
        "arn:aws:ec2:*:*:snapshot/*",
        "arn:aws:ec2:*:*:image/*"
    ],
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction": [
                "CreateImage",
                "CreateSnapshots"
            ]
        }
    }
},

```

```

    {
      "Sid": "CreateTagsAfterResourceCreation",
      "Effect": "Allow",
      "Action": "ec2:CreateTags",
      "Resource": [
        "arn:aws:ec2:*:*:snapshot/*",
        "arn:aws:ec2:*:*:image/*"
      ],
      "Condition": {
        "StringLike": {
          "ec2:ResourceTag/AwsVssConfig": "*"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": [
            "AppConsistent",
            "Device"
          ]
        }
      }
    },
    {
      "Sid": "DescribeImagesAndSnapshots",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
    }
  ]
}

```

Merampingkan izin untuk kasus penggunaan tertentu (lanjutan)

Kebijakan `AWSEC2VssSnapshotPolicy` terkelola menyertakan izin untuk semua cara Anda dapat membuat snapshot berkemampuan 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 berkemampuan VSS hanya melalui AWS Backup layanan, maka Anda dapat merampingkan pernyataan kebijakan sebagai berikut.

- Hilangkan pernyataan kebijakan yang diidentifikasi oleh ID pernyataan berikut (SID):
 - `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.

- Hilangkan pernyataan kebijakan yang diidentifikasi oleh ID pernyataan berikut (SID):
 - `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.

Kebijakan lama (tidak lagi didukung)

Kebijakan lama yang memberikan izin untuk snapshot berkemampuan VSS mencakup izin IAM yang direkomendasikan sebelum rilis kebijakan terkelola. `AWSEC2VssSnapshotPolicy`

Jika Anda telah mengonfigurasi peran instance dengan kebijakan lama, Anda dapat terus menggunakannya. Namun, untuk memastikan bahwa kebijakan Anda tetap terkini dengan praktik terbaik IAM terbaru dan cakupan pernyataan kebijakan yang sesuai, kami sarankan Anda mengganti kebijakan lama dengan kebijakan terkelola. `AWSEC2VssSnapshotPolicy`

Contoh kebijakan

Contoh kebijakan berikut menggunakan `ec2:DescribeInstanceAttribute` yang didukung dalam `AwsVssComponents` paket versi 2.2.1 dan yang lebih baru. Jika Anda memiliki versi `AwsVssComponents` paket yang lebih lama diinstal, Anda harus menggantinya dengan `ec2:DescribeInstances` tindakan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CreateTags",
      "Resource": [
        "arn:aws:ec2:*::snapshot/*",
        "arn:aws:ec2:*::image/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstanceAttribute",
        "ec2:CreateSnapshot",
        "ec2:CreateSnapshots",
        "ec2:CreateImage",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
    }
  ]
}
```

Untuk informasi selengkapnya tentang kebijakan terkelola IAM, lihat [kebijakan AWS terkelola](#) di Panduan Pengguna IAM.

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.

Ada beberapa cara untuk menginstal komponen ke instans yang sudah ada:

- (Direkomendasikan) [Jalankan dokumen `VssInstallAndSnapshot` perintah `AWSEC 2` \(disarankan\)](#). Operasi ini secara otomatis menginstal atau memperbarui jika diperlukan setiap kali dijalankan.
- [Instal komponen VSS secara manual pada sebuah instans](#).
- [Perbarui komponen VSS pada instans Anda sesuai jadwal](#).

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 sebuah instans](#)
- [Perbarui komponen VSS pada instans Anda sesuai jadwal](#)

Instal komponen VSS secara manual pada sebuah instans

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 instalasi manual, kami sarankan Anda menggunakan paket komponen AWS VSS terbaru untuk meningkatkan keandalan dan kinerja snapshot yang konsisten 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 [Jalankan dokumen VssInstallAndSnapshot perintah AWSEC 2 \(disarankan\)](#).

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 Command, pilih tombol di sebelah `AWS-configure AWSPackage`.
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 [Buat Profil Instans IAM untuk Systems Manager](#) dalam Panduan Pengguna AWS Systems Manager .

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 dengan menggunakan AWS CLI

- 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-01234567890abcdef" \  
  --parameters '{"action":["Install"],"name":["AwsVssComponents"]}'
```

PowerShell

Gunakan prosedur berikut untuk mengunduh dan menginstal `AwsVssComponents` paket pada instance Anda dengan menggunakan Run Command dari Tools for 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 menggunakan AWS Tools for Windows PowerShell

- Jalankan perintah berikut untuk mengunduh dan menginstal komponen VSS yang diperlukan untuk System Manager.

```
Send-SSMCommand -DocumentName AWS-ConfigureAWSPackage -InstanceId  
"i-01234567890abcdef" -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](#).
2. Arahkan ke `C:\Program Files\Amazon\AwsVss Components`.
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 penandatanganan adalah Amazon Web Services Inc.
5. Gunakan langkah-langkah sebelumnya untuk memverifikasi tanda tangan pada `Ec2VssInstaller` dan `Ec2VssProvider.dll`.

Perbarui komponen VSS pada instans Anda sesuai jadwal

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 sebuah instans](#) 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 [Jalankan dokumen VssInstallAndSnapshot perintah AWSEC 2 \(disarankan\)](#).

Untuk membuat kaitan Systems Manager State Manager, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Untuk membuat asosiasi State Manager menggunakan konsol

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. Dalam daftar Dokumen, pilih `AWS-configure AWSPackage`.
6. Di bagian Parameter, pilih Instal dari daftar Tindakan.
7. Untuk Jenis penginstalan, pilih Hapus instalasi dan instal ulang.
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 [Tentang kepatuhan kaitan State Manager](#). 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 instance, dan verifikasi bahwa instance dikonfigurasi dengan peran AWS Identity and Access Management (IAM) untuk Systems Manager. Untuk informasi selengkapnya, lihat [Menyiapkan AWS Systems Manager](#).

AWS CLI

Anda dapat menjalankan AWS CLI 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 Manajer Negara menggunakan AWS CLI

1. Instal dan konfigurasi AWS CLI, jika Anda belum melakukannya. Untuk informasi, lihat [Menginstal atau memperbarui versi terbaru dari AWS CLI](#).
2. 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-01234567890abcdef\  
\ "i-000011112222abcde\"}]]
```

Untuk informasi tentang opsi lain yang dapat Anda gunakan dengan `create-association` perintah, lihat [create-association](#) di AWS Systems Manager bagian Referensi Perintah. AWS CLI

Buat snapshot EBS yang didukung VSS

Bagian ini mencakup langkah-langkah untuk membuat snapshot EBS yang didukung VSS.

Anda dapat membuat snapshot EBS yang didukung VSS dari volume EBS yang dilampirkan ke instans EC2 Anda. Sebelum Anda mencoba membuat snapshot yang didukung VSS, pastikan [Prasyarat](#) sudah terpenuhi.

Topik

- [Buat snapshot VSS dengan dokumen perintah AWS Systems Manager](#)
- [Buat snapshot VSS menggunakan AWS Backup](#)
- [Membuat snapshot VSS menggunakan Amazon Data Lifecycle Manager](#)

Buat snapshot VSS dengan dokumen perintah AWS Systems Manager

Anda dapat menggunakan dokumen AWS Systems Manager perintah untuk membuat snapshot berkemampuan 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](#).

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:

ExcludeBootVolume (string, opsional)

Pengaturan ini mengecualikan volume boot dari proses pencadangan jika Anda membuat snapshot. Untuk mengecualikan volume boot dari snapshot Anda, atur 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`.

CopyOnly(string, opsional)

Jika Anda menggunakan cadangan SQL Server asli selain AWS VSS, melakukan pencadangan khusus Salin mencegah AWS VSS memutus rantai cadangan diferensial asli. Untuk melakukan operasi pencadangan hanya-salin, 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) yang didukung VSS untuk mencadangkan instans Anda, atur 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 AMI yang didukung Amazon EBS-Backed](#).

AmiName(string, opsional)

Jika `CreateAmi` disetel ke `True`, tentukan nama AMI yang dibuat cadangan.

description (string, opsional)

Tentukan deskripsi untuk snapshot atau gambar yang dibuat proses ini.

tanda (string, opsional)

Kami menyarankan Anda menandai snapshot dan gambar Anda untuk membantu Anda menemukan dan mengelola sumber daya Anda, misalnya, untuk 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`.

Secara default, sistem menambahkan tag cadangan berikut untuk snapshot dan gambar berkemampuan VSS.

- **Perangkat** — Untuk snapshot berkemampuan VSS, ini adalah nama perangkat dari volume EBS yang ditangkap snapshot.
- **AppConsistent**— Tag ini menunjukkan keberhasilan pembuatan snapshot berkemampuan VSS atau AMI.
- **AwsVssConfig** — Ini mengidentifikasi snapshot dan AMI yang dibuat dengan VSS diaktifkan. Tag mencakup informasi meta seperti `AwsVssComponents` versi.

Warning

Menentukan salah satu tag cadangan ini dalam daftar parameter Anda akan menyebabkan kesalahan.

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.

CollectDiagnosticLog (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 terkonsolidasi disimpan sebagai arsip `.zip` format di lokasi berikut pada instans Anda:

C:\ProgramData\Amazon\AwsVss\Logs*timestamp*.zip

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 melewati 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 berkemampuan VSS dengan AWS Systems Manager dokumen perintah sebagai berikut.

Jalankan dokumen VssInstallAndSnapshot perintah AWSEC 2 (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 yang didukung 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. Konfigurasi 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 di AWS CLI untuk membuat snapshot EBS berkemampuan VSS dan mendapatkan status pembuatan snapshot Anda.

Buat snapshot EBS yang didukung VSS

Jalankan perintah berikut untuk membuat snapshot EBS yang didukung 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 berkemampuan VSS dan dapatkan status runtime saat ini untuk pembuatan output Anda. Tentukan parameter yang dijelaskan dalam daftar sebelumnya untuk mengubah perilaku proses snapshot.

Buat snapshot EBS berkemampuan VSS dengan Tools untuk Windows PowerShell

Jalankan perintah berikut untuk membuat snapshot EBS yang didukung 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 CreateVssSnapshot perintah AWSEC 2

Untuk menjalankan dokumen AWSEC2-CreateVssSnapshot, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Buat snapshot EBS yang didukung 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. Konfigurasi 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.

- Untuk parameter tambahan guna menentukan perilaku Run Command Systems Manager seperti Kontrol laju, masukkan nilai seperti yang dijelaskan dalam [Menjalankan perintah dari konsol](#).
- 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 di AWS CLI untuk membuat snapshot EBS berkemampuan VSS.

Buat snapshot EBS yang didukung VSS

Jalankan perintah berikut untuk membuat snapshot EBS yang didukung 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 berkemampuan VSS.

Buat snapshot EBS berkemampuan VSS dengan Tools untuk Windows PowerShell

Jalankan perintah berikut untuk membuat snapshot EBS yang didukung 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 Kluster Failover Windows dengan penyimpanan EBS bersama

Anda dapat menggunakan salah satu prosedur baris perintah yang dijelaskan di bagian sebelumnya untuk membuat snapshot yang didukung VSS. Dokumen perintah (`AWSEC2-VssInstallAndSnapshot` atau `AWSEC2-CreateVssSnapshot`) harus berjalan pada simpul primer di kluster Anda. Dokumen akan gagal pada simpul sekunder karena tidak memiliki akses ke disk bersama. Jika primer dan sekunder Anda 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.

Jalankan dokumen perintah AWSEC 2- ManageVss IO SSM

Anda dapat menggunakan skrip berikut dan dokumen SSM AWSEC2-`ManageVssIO` yang ditentukan sebelumnya untuk menghentikan sementara I/O, membuat snapshot EBS yang didukung VSS, dan memulai ulang I/O. Proses ini berjalan dalam konteks pengguna yang menjalankan perintah. Jika pengguna memiliki izin yang cukup untuk membuat dan menandai snapshot, maka AWS Systems Manager dapat membuat dan menandai snapshot EBS berkemampuan VSS tanpa perlu peran snapshot IAM tambahan pada instance.

Sebaliknya, dokumen perintah (`AWSEC2-VssInstallAndSnapshot` atau `AWSEC2-CreateVssSnapshot`) mengharuskan Anda menetapkan peran snapshot IAM ke setiap instans yang ingin Anda buat snapshot EBS. Jika Anda tidak ingin memberikan izin IAM tambahan untuk instans Anda karena alasan kebijakan atau kepatuhan, Anda dapat menggunakan skrip berikut.

Sebelum Anda memulai

Perhatikan detail penting berikut tentang proses ini:

- Proses ini menggunakan PowerShell script (`CreateVssSnapshotAdvancedScript.ps1`) untuk mengambil snapshot dari semua volume pada instance yang Anda tentukan, kecuali volume root. Jika Anda perlu mengambil snapshot volume root, Anda harus menggunakan dokumen SSM `AWSEC2-CreateVssSnapshot`.
- Skripnya memanggil dokumen `AWSEC2-ManageVssIO` dua kali. Pertama kali dengan parameter `Action` diatur ke `Freeze`, yang menjeda semua I/O pada instans. Kedua kalinya, `Action` parameter diatur menjadi `Thaw`, yang memaksa I/O untuk melanjutkan.
- Jangan mencoba menggunakan `AWSEC2-ManageVssIO` dokumen tanpa menggunakan `CreateVssSnapshotAdvancedScript` skrip.ps1. Kerangka VSS Microsoft mensyaratkan bahwa tindakan `Freeze` dan `Thaw` dipanggil tidak lebih dari sepuluh detik, dan memanggil tindakan ini secara manual tanpa skrip dapat mengakibatkan kesalahan.

Untuk membuat snapshot EBS yang didukung VSS menggunakan dokumen SSM **AWSEC2-`ManageVssIO`**

1. Unduh file [CreateVssSnapshotAdvancedScript.zip](#) dan ekstrak konten file.
2. Buka `CreateVssSnapshotAdvancedScript.ps1` di editor teks, edit panggilan sampel di bagian bawah skrip dengan ID instans EC2 yang valid, deskripsi snapshot, dan nilai tag yang diinginkan, lalu jalankan skrip dari PowerShell

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 tersebut dalam daftar volume EBS.

Note

Untuk mengotomatiskan backup, Anda dapat membuat tugas jendela AWS Systems Manager pemeliharaan yang menggunakan dokumen `AWSEC2-VssInstallAndSnapshot` Untuk informasi selengkapnya, lihat [Bekerja dengan Jendela Pemeliharaan \(Konsol\)](#) dalam Panduan Pengguna AWS Systems Manager .

Buat snapshot VSS menggunakan AWS Backup

Anda dapat membuat cadangan VSS saat menggunakan AWS Backup dengan mengaktifkan VSS di konsol atau CLI. Pastikan Anda telah memenuhi [prasyarat](#) sebelum membuat paket cadangan dengan VSS aktif. Untuk informasi selengkapnya, lihat [Membuat cadangan VSS Windows](#) di Panduan Developer AWS Backup .

Note

AWS Backup tidak secara otomatis menginstal `AwsVssComponents` paket pada instance Anda. Anda harus melakukan instalasi manual pada instans. Untuk informasi selengkapnya, lihat [Instal komponen VSS secara manual pada sebuah instans](#).

Membuat snapshot VSS menggunakan 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 <https://docs.aws.amazon.com/ebs/latest/userguide/automate-app-consistent-backups.html>.

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 sebuah instans](#).

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](#).
- 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 instance 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 galat saat membuat snapshot EBS berkemampuan VSS, Anda dapat melihat output perintah di konsol Systems Manager.

Untuk dokumen Systems Manager yang membuat snapshot VSS, Anda dapat mengatur `CollectDiagnosticLogs` parameter ke "True" saat runtime. Ketika `CollectDiagnosticLogs` parameter diatur ke "True", VSS mengumpulkan log tambahan untuk membantu dalam debugging. Untuk informasi selengkapnya, lihat [Kumpulkan log diagnostik tambahan](#).

Jika Anda mengumpulkan log diagnostik, dokumen Systems Manager menyimpannya di instans 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 AWS Support.

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 Systems Manager dengan titik akhir VPC, dan tindakan Systems Manager [SendCommand](#)API (Jalankan Perintah di konsol) gagal, verifikasi bahwa Anda mengonfigurasi titik akhir berikut dengan benar: `com.amazonaws.wilayah.ec2`.

Tanpa titik akhir Amazon EC2 ditentukan, panggilan untuk menghitung volume EBS terlampir gagal, yang menyebabkan perintah Systems Manager 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 Systems Manager 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 AWSEC2-CreateVssSnapshot Systems Manager 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 AWSEC2-CreateVssSnapshot Systems Manager 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 instance dengan proxy yang dikonfigurasi

Jika Anda mengalami masalah saat membuat snapshot EBS berkemampuan VSS pada instans yang menggunakan proksi untuk mencapai titik akhir EC2, pastikan hal berikut:

- Proxy dikonfigurasi sehingga titik akhir layanan EC2 di Wilayah instans dan IMDS dapat dijangkau dengan menjalankan sebagai SYSTEM. AWS Tools for Windows PowerShell
- AwsVssComponents versi 2.0.1 atau yang lebih baru diinstal. Dimulai dengan AwsVssComponents versi 2.0.1, penyedia EC2 VSS mendukung penggunaan proksi WinHTTP yang dikonfigurasi sistem. 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

EC2 Windows VSS Provider mungkin kehabisan waktu karena aktivitas atau layanan pada instans mencegah snapshot dengan dukungan VSS melanjutkan pada waktu yang tepat. 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:

- I/O berlebihan untuk 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 konfigurasi 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 throttling, gunakan paket `AwsVssComponents` terbaru (versi 2.1.0 dan di atasnya, dengan izin prasyarat). Paket ini memanfaatkan tindakan API `CreateSnapshots` EC2 untuk mengurangi jumlah tindakan mutasi seperti pembuatan dan penandaan snapshot per volume.
- 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 tingkat Systems Manager untuk mengurangi konkurensi untuk layanan seperti AWS Backup itu menggunakan Systems Manager 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.
- Jika Anda menggunakan tipe instans yang lebih kecil seperti `t2` / `t3` / `t3a.nano` atau `t2` / `t3` / `t3a.micro`, waktu habis karena memori dan kendala 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`. Dokumen `AWSEC2-ManageVssIo` SSM `AWSEC2-CreateVssSnapshot` dan SSM harus dikonfigurasi ke `FullLanguage` mode. PowerShell

Untuk memverifikasi mode bahasa, jalankan perintah berikut pada instance di PowerShell konsol:

```
$ExecutionContext.SessionState.LanguageMode
```

Untuk informasi selengkapnya, lihat [about_Language_Modes](#) di dokumentasi Microsoft.

Pulihkan volume EBS dari snapshot EBS yang mendukung VSS

Anda dapat menggunakan `RestoreVssSnapshotSampleScript.ps1` untuk mengembalikan volume pada sebuah instans dari snapshot EBS yang mendukung VSS. Skrip ini 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.

Pulihkan volume EBS dari snapshot EBS yang mendukung VSS

1. Unduh file [RestoreVssSnapshotSampleScript.zip](#) dan ekstrak konten file.
2. Buka `RestoreVssSnapshotSampleScript.ps1` di editor teks dan edit panggilan sampel di bagian bawah skrip dengan ID instans EC2 yang valid dan ID snapshot EBS, lalu jalankan skrip dari PowerShell

AWS riwayat versi VSS solution

Topik

- [AwsVssComponents versi paket](#)
- [Dukungan versi Windows OS](#)

AwsVssComponents versi paket

Tabel berikut menjelaskan versi yang dirilis dari paket komponen AWS VSS.

Versi	Detail	Tanggal rilis
2.3.2		9 Mei 2024

Versi	Detail	Tanggal rilis
	Memperbaiki kasus di mana pendaftaran penyedia VSS tidak dihapus saat penghapusan instalasi.	
2.3.1	Menambahkan tag default baru <code>AwsVssConfig</code> untuk mengidentifikasi snapshot dan AMI yang dibuat oleh AWS VSS.	7 Maret 2024
2.2.1	<ul style="list-style-type: none">• Menambahkan dukungan untuk menggunakan <code>DescribeInstanceAttribute</code> API.• Perbaiki bug dan peningkatan keandalan.• Dukungan usang untuk Windows Server 2012 dan 2012 R2. AWS Komponen VSS versi 2.2.1 instalasi pada Windows Server 2012 dan 2012 R2 akan gagal. AWS Komponen VSS versi 2.1.0 adalah versi terakhir yang mendukung Windows Server 2012 dan 2012 R2.	Januari 18, 2024
2.1.0	Menambahkan dukungan untuk menggunakan <code>CreateSnapshots</code> API.	6 November 2023
2.0.1	Dukungan tambahan untuk menggunakan pengaturan proksi WinHTTP.	26 Oktober 2023
2.0.0	Menambahkan kemampuan ke komponen AWS VSS untuk membuat snapshot dan AMI, yang memungkinkan kompatibilitas dengan logging PowerShell modul, logging blok skrip, dan fitur transkripsi.	28 April 2023
1.3.2.0	Memperbaiki kasus di mana kegagalan instalasi tidak dilaporkan dengan benar.	10 Mei 2022

Versi	Detail	Tanggal rilis
1.3.1.0	<ul style="list-style-type: none"> • Snapshot tetap gagal di pengendali domain dalam kaitannya dengan kesalahan logging penulis NTDS VSS. • Kesalahan agen VSS tetap saat menghapus pemasangan penyedia VSS versi 1.0. 	6 Februari 2020
1.3.00	<ul style="list-style-type: none"> • Penebangan yang lebih baik dengan mengurangi kata benda yang tidak diinginkan. • Masalah wilayah diperbaiki selama instalasi. • Memperbaiki kode pengembalian untuk beberapa kondisi kesalahan pendaftaran penyedia. • Memperbaiki berbagai masalah instalasi. 	19 Maret 2019
1.2.00	<ul style="list-style-type: none"> • Menambahkan parameter baris perintah <code>-nw</code> (tidak-menulis) dan <code>-copy</code> (hanya-salinan) kepada agen. • Memperbaiki EventLog kesalahan yang disebabkan oleh panggilan alokasi memori yang tidak tepat. 	15 November 2018
1.1	Memperbaiki komponen AWS VSS yang digunakan secara tidak benar sebagai penyedia Backup dan Restore Windows default.	12 Desember 2017
1.0	Rilis awal.	20 November 2017

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-nama VssInstal IAndSnaps hot versi	AWSEC2-nama CreateVss Snapshot versi	AWSEC2-Nama versi ManageVss IO
Windows Server 2022	default	default	default	default
Windows Server 2019	default	default	default	default
Windows Server 2016	default	default	default	default
Windows Server 2012 R2	2.1.0	Tidak didukung	2012R2	2012R2
Windows Server 2012	2.1.0	Tidak didukung	2012R2	2012R2
Windows Server 2008 R2	1.3.1.0	Tidak didukung	2008R2	2008R2

Pencegahan penulisan robek untuk instance Linux

Note

Pencegahan penulisan robek hanya didukung dengan instance Linux.

Pencegahan penulisan robek adalah fitur penyimpanan blok yang dirancang AWS untuk meningkatkan kinerja beban kerja database relasional intensif I/O 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 menuliskannya 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. Overhead I/O tambahan yang terkait dengan penulisan ke buffer penulisan ganda berdampak pada performa basis data 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.

Ukuran blok yang didukung dan perataan batas blok

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 volume penyimpanan instans NVMe dilampirkan ke instans i-family generasi saat ini.	Instans i4i, iM4GN, dan IS4gen didukung oleh Nitro SSD. AWS	
Volume Amazon EBS	Semua volume Amazon EBS yang dilampirkan ke instans yang dibangun di Sistem AWS Nitro .		

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 dan konfigurasi pencegahan tumpang tindih](#).

Persyaratan

Agar pencegahan tumpang tindih berfungsi dengan baik, operasi I/O harus memenuhi persyaratan ukuran, keselarasan, dan batas, sebagaimana ditentukan dalam bidang NTWPU, NTWGU, NTWBU. Anda harus mengonfigurasi sistem operasi untuk memastikan bahwa subsistem penyimpanan tertentu (sistem file, LVM, RAID, dll.) tidak memodifikasi properti I/O pada tumpukan penyimpanan, termasuk penggabungan blok, pemisahan, atau relokasi alamat blok, sebelum dikirimkan ke perangkat.

Pencegahan tumpang tindih 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 penggabungan I/O untuk beban kerja MySQL dan MariaDB.

Periksa dukungan dan konfigurasi pencegahan tumpang tindih

Untuk mengonfirmasi apakah instans dan volume Anda mendukung pencegahan tumpang tindih, dan untuk melihat data spesifik vendor ruang nama NVMe yang berisi informasi pencegahan tumpang tindih, gunakan perintah berikut ini.

```
$ 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 ruang nama NVMe yang berisi informasi pencegahan tumpang tindih untuk file `/dev/nvme1n1`.

```
$ sudo nvme id-ns -v /dev/nvme1n1
```

Jika instans dan volume Anda mendukung pencegahan penulisan yang robek, ini mengembalikan informasi pencegahan penulisan AWS robek berikut dalam data spesifik vendor namespace NVMe.

Note

Byte pada tabel berikut ini mewakili offset dalam byte dari awal data spesifik vendor ruang nama NVMe.

Byte	Deskripsi
0:31	Nama titik pemasangan lampiran perangkat, misalnya <code>/dev/xvda</code> . Anda memberikan ini selama permintaan lampiran volume dan dapat digunakan oleh instans Amazon EC2 untuk membuat symlink ke perangkat blok NVMe (<code>nvmeXn1</code>).

Byte	Deskripsi
32:63	ID volume. Misalnya, <code>vo101234567890abcdef</code> . 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 $NTWPG * n \leq NTWPU$, 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 <code>0h</code> 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.

Konfigurasi tumpukan perangkat lunak Anda untuk pencegahan tumpang tindih

Pencegahan tumpang tindih diaktifkan secara default pada [tipe instans yang didukung dengan volume yang didukung](#). Anda tidak perlu mengaktifkan pengaturan tambahan apa pun untuk mengaktifkan volume atau instans Anda untuk pencegahan tumpang tindih.

Note

Tidak ada dampak performa pada beban kerja yang tidak mendukung pencegahan tumpang tindih. Anda tidak perlu melakukan perubahan apa pun untuk beban kerja ini.

Beban kerja yang mendukung pencegahan tumpang tindih, tetapi tidak dikonfigurasi untuk menggunakannya, terus menggunakan buffer penulisan ganda dan tidak menerima manfaat performa 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. Konfigurasi volume Anda untuk menggunakan sistem file ext4 dengan BigAlloc opsi dan atur ukuran cluster ke 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. Konfigurasi InnoDB untuk menggunakan metode flush `0_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=0_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 superbloc 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](#).

Sumber daya dan tanda

Amazon EC2 menyediakan berbagai sumber daya yang dapat Anda buat dan gunakan. Beberapa dari sumber daya ini mencakup citra, instans, volume, dan snapshot. Saat Anda membuat sebuah sumber daya, kami menetapkan sebuah ID sumber daya yang unik untuk sumber daya tersebut.

Beberapa sumber daya dapat ditandai dengan nilai yang Anda tentukan untuk membantu mengatur dan mengidentifikasinya.

Topik berikut ini menjelaskan sumber daya dan tanda, dan cara Anda menggunakannya.

Daftar Isi

- [Keranjang Sampah](#)
- [Lokasi sumber daya](#)
- [ID sumber daya](#)
- [Membuat daftar dan memfilter sumber daya Anda](#)
- [Amazon EC2 Global View](#)
- [Tandai sumber daya Amazon EC2 Anda](#)
- [Kuota layanan Amazon EC2](#)

Keranjang Sampah

Keranjang Sampah adalah fitur pemulihan data yang memungkinkan Anda memulihkan snapshot dan AMI yang didukung Amazon EBS yang terhapus secara tidak sengaja. Saat menggunakan Keranjang Sampah, jika sumber daya Anda dihapus, sumber daya tersebut dipertahankan di Keranjang Sampah untuk jangka waktu yang Anda tentukan sebelum dihapus secara permanen.

Anda dapat memulihkan sumber daya dari Keranjang Sampah kapan saja sebelum periode retensi berakhir. Setelah memulihkan sumber daya dari Keranjang Sampah, sumber daya tersebut akan dihapus dari Keranjang Sampah dan Anda dapat menggunakannya dengan cara yang sama seperti menggunakan sumber daya lain dari tipe tersebut di akun Anda. Jika periode retensi berakhir dan sumber daya tidak dipulihkan, sumber daya tersebut akan dihapus secara permanen dari Keranjang Sampah dan tidak lagi tersedia untuk pemulihan.

Menggunakan Keranjang Sampah membantu memastikan kelangsungan bisnis dengan melindungi data penting bisnis Anda dari penghapusan yang tidak disengaja.

Topik

- [Bagaimana cara kerjanya?](#)
- [Sumber daya yang didukung](#)
- [Pertimbangan](#)
- [Kuota](#)
- [Layanan-layanan terkait](#)
- [Harga](#)
- [Izin IAM yang diperlukan](#)
- [Bekerja dengan aturan retensi](#)
- [Bekerja dengan sumber daya di Keranjang Sampah](#)
- [Pantau Keranjang Sampah](#)

Bagaimana cara kerjanya?

Untuk mengaktifkan dan menggunakan Recycle Bin, Anda harus membuat aturan retensi di AWS Wilayah tempat Anda ingin melindungi sumber daya Anda. Aturan retensi menentukan hal berikut:

- Tipe sumber daya yang ingin Anda lindungi.
- Sumber daya yang ingin Anda pertahankan di Keranjang Sampah saat dihapus.
- Periode retensi untuk mempertahankan sumber daya di Keranjang Sampah sebelum dihapus secara permanen.

Dengan Keranjang Sampah, Anda dapat membuat dua tipe aturan retensi:

- Aturan retensi tingkat tanda — Aturan retensi tingkat tanda menggunakan tanda sumber daya untuk mengidentifikasi sumber daya yang akan dipertahankan di Keranjang Sampah. Untuk setiap aturan retensi, Anda menentukan satu atau beberapa pasangan kunci dan nilai tanda. Sumber daya tipe tertentu yang ditandai dengan setidaknya satu pasangan kunci dan nilai tanda yang ditentukan dalam aturan retensi secara otomatis dipertahankan di Keranjang Sampah setelah penghapusan. Gunakan tipe aturan retensi ini jika ingin melindungi sumber daya tertentu di akun Anda berdasarkan tandanya.
- Aturan retensi tingkat wilayah — Aturan retensi tingkat wilayah tidak memiliki tanda sumber daya yang ditentukan. Aturan ini berlaku untuk semua sumber daya dari tipe tertentu di Wilayah tempat

aturan dibuat, bahkan jika sumber daya tidak ditandai. Gunakan tipe aturan retensi ini jika Anda ingin melindungi semua sumber daya tipe tertentu di Wilayah tertentu.

Sementara sumber daya berada di Keranjang Sampah, Anda memiliki kemampuan untuk mengembalikan sumber daya tersebut agar dapat digunakan kapan saja.

Sumber daya tetap berada di Keranjang Sampah sampai salah satu hal berikut terjadi:

- Anda mengembalikannya secara manual untuk digunakan. Saat Anda memulihkan sumber daya dari Keranjang Sampah, sumber daya tersebut dihapus dari Keranjang Sampah dan segera tersedia untuk digunakan. Anda dapat menggunakan sumber daya yang dipulihkan dengan cara yang sama seperti sumber daya lain dari tipe tersebut pada akun
- Periode retensi berakhir. Jika periode retensi berakhir dan sumber daya belum dipulihkan dari Keranjang Sampah, sumber daya tersebut dihapus secara permanen dari Keranjang Sampah serta tidak dapat lagi dilihat atau dipulihkan.

Sumber daya yang didukung

Keranjang Sampah mendukung tipe sumber daya berikut:

- Snapshot Amazon EBS

Important

Aturan retensi Keranjang Sampah juga berlaku untuk snapshot yang diarsipkan pada tingkat penyimpanan arsip. Jika Anda menghapus snapshot yang diarsipkan yang cocok dengan aturan retensi, snapshot tersebut akan dipertahankan di Keranjang Sampah untuk periode yang ditentukan dalam aturan retensi. Snapshot yang diarsipkan dikenai biaya dengan tarif untuk snapshot yang diarsipkan saat berada di Keranjang Sampah.

- Amazon Machine Images (AMI) yang didukung Amazon EBS

Note

Aturan retensi juga berlaku untuk AMI yang dinonaktifkan.

Pertimbangan

Pertimbangan berikut berlaku saat bekerja dengan Keranjang Sampah dan aturan retensi.

Pertimbangan umum

-  **Important**
Saat membuat aturan retensi pertama Anda, dibutuhkan waktu hingga 30 menit agar aturan tersebut aktif dan mulai mempertahankan sumber daya. Setelah Anda membuat aturan retensi pertama, aturan retensi berikutnya menjadi aktif dan hampir secara langsung mulai mempertahankan sumber daya.
- Jika sumber daya cocok dengan lebih dari satu aturan retensi setelah penghapusan, aturan retensi dengan periode retensi terpanjang akan diutamakan.
- Anda tidak dapat menghapus sumber daya secara manual dari Keranjang Sampah. Sumber daya akan dihapus secara otomatis saat periode retensi berakhir.
- Saat sumber daya ada di Keranjang Sampah, Anda hanya dapat melihatnya, memulihkannya, atau memodifikasi tandanya. Untuk menggunakan sumber daya dengan cara lain, Anda harus memulihkannya terlebih dahulu.
- Jika ada Layanan AWS, seperti AWS Backup atau Amazon Data Lifecycle Manager, menghapus sumber daya yang cocok dengan aturan retensi, sumber daya tersebut secara otomatis disimpan oleh Recycle Bin.
- Ketika sumber daya dikirim ke Keranjang Sampah, tanda yang dihasilkan oleh sistem berikut ditetapkan ke sumber daya:
 - Kunci tanda — `aws:recycle-bin:resource-in-bin`
 - Nilai tanda — `true`

Anda tidak dapat mengedit atau menghapus tanda ini secara manual. Ketika sumber daya dipulihkan dari Keranjang Sampah, tanda secara otomatis dihapus.

Pertimbangan untuk snapshot

-  **Important**
Jika Anda memiliki aturan retensi untuk AMI dan snapshot yang terkait dengan AMI tersebut, buat periode retensi snapshot sama dengan atau lebih lama dari periode retensi

AMI. Hal ini memastikan bahwa Keranjang Sampah tidak menghapus snapshot yang terkait dengan AMI sebelum menghapus AMI itu sendiri, karena ini akan membuat AMI tidak dapat dipulihkan.

- Jika snapshot diaktifkan untuk pemulihan snapshot cepat saat dihapus, pemulihan snapshot cepat dinonaktifkan secara otomatis segera setelah snapshot dikirim ke Keranjang Sampah.
 - Jika Anda memulihkan snapshot sebelum pemulihan snapshot cepat dinonaktifkan untuk snapshot tersebut, snapshot tersebut tetap diaktifkan.
 - Jika Anda mengembalikan snapshot, setelah pemulihan snapshot cepat dinonaktifkan, snapshot tersebut tetap dinonaktifkan. Jika perlu, Anda harus mengaktifkan kembali pemulihan snapshot cepat secara manual.
- Jika snapshot dibagikan saat dihapus, snapshot tersebut secara otomatis tidak dibagikan saat dikirim ke Keranjang Sampah. Jika Anda memulihkan snapshot, semua izin berbagi sebelumnya secara otomatis dipulihkan.
- Jika snapshot yang dibuat oleh AWS layanan lain, seperti AWS Backup dikirim ke Recycle Bin dan Anda kemudian mengembalikan snapshot itu dari Recycle Bin, itu tidak lagi dikelola oleh AWS layanan yang membuatnya. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.

Pertimbangan untuk AMI

- Hanya AMI yang didukung Amazon EBS yang didukung.

Important

Jika Anda memiliki aturan retensi untuk AMI dan snapshot yang terkait dengan AMI tersebut, buat periode retensi snapshot sama dengan atau lebih lama dari periode retensi AMI. Hal ini memastikan bahwa Keranjang Sampah tidak menghapus snapshot yang terkait dengan AMI sebelum menghapus AMI itu sendiri, karena ini akan membuat AMI tidak dapat dipulihkan.

- Jika AMI dibagikan saat dihapus, AMI tersebut secara otomatis tidak dibagikan saat dikirim ke Keranjang Sampah. Jika Anda memulihkan AMI, semua izin berbagi sebelumnya akan dipulihkan secara otomatis.

- Sebelum dapat memulihkan AMI dari Keranjang Sampah, Anda harus terlebih dahulu memulihkan semua snapshot yang terkait dari Keranjang Sampah dan memastikan bahwa snapshot tersebut berada dalam status `available`.
- Jika snapshot yang terkait dengan AMI dihapus dari Keranjang Sampah, AMI tersebut tidak lagi dapat dipulihkan. AMI akan dihapus saat periode retensi berakhir.
- Jika AMI yang dibuat oleh AWS layanan lain, seperti AWS Backup, dikirim ke Recycle Bin dan Anda kemudian mengembalikan AMI itu dari Recycle Bin, itu tidak lagi dikelola oleh AWS layanan yang membuatnya. Anda harus menghapus AMI secara manual jika tidak lagi diperlukan.

Pertimbangan untuk kebijakan snapshot Amazon Data Lifecycle Manager

- Jika Amazon Data Lifecycle Manager menghapus snapshot yang cocok dengan aturan retensi, snapshot tersebut secara otomatis dipertahankan oleh Keranjang Sampah.
- Jika Amazon Data Lifecycle Manager menghapus snapshot dan mengirimkannya ke Keranjang Sampah saat ambang batas retensi kebijakan tercapai, serta Anda memulihkan snapshot tersebut dari Keranjang Sampah secara manual, Anda harus menghapus snapshot tersebut secara manual saat tidak lagi diperlukan. Amazon Data Lifecycle Manager tidak akan lagi mengelola snapshot.
- Jika Anda menghapus snapshot yang dibuat oleh kebijakan secara manual, dan snapshot tersebut ada di Keranjang Sampah saat ambang penyimpanan kebijakan tercapai, Amazon Data Lifecycle Manager tidak akan menghapus snapshot tersebut. Amazon Data Lifecycle Manager tidak mengelola snapshot saat disimpan di Keranjang Sampah.

Jika snapshot dipulihkan dari Keranjang Sampah sebelum ambang retensi kebijakan tercapai, Amazon Data Lifecycle Manager akan menghapus snapshot tersebut saat ambang retensi kebijakan tercapai.

Jika snapshot dipulihkan dari Keranjang Sampah setelah ambang batas retensi kebijakan tercapai, Amazon Data Lifecycle Manager tidak akan lagi menghapus snapshot tersebut. Anda harus menghapus snapshot secara manual saat tidak lagi diperlukan.

Pertimbangan untuk Backup AWS

- Jika AWS Backup menghapus snapshot yang cocok dengan aturan retensi, snapshot tersebut secara otomatis disimpan oleh Recycle Bin.

Pertimbangan untuk snapshot yang diarsipkan

- Aturan retensi Keranjang Sampah juga berlaku untuk snapshot yang diarsipkan pada tingkat penyimpanan arsip. Jika Anda menghapus snapshot yang diarsipkan yang cocok dengan aturan retensi, snapshot tersebut akan dipertahankan di Keranjang Sampah untuk periode yang ditentukan dalam aturan retensi.

Snapshot yang diarsipkan dikenai biaya dengan tarif untuk snapshot yang diarsipkan saat berada di Keranjang Sampah.

Jika aturan retensi menghapus snapshot yang diarsipkan dari Keranjang Sampah sebelum periode arsip minimum 90 hari, Anda akan dikenai biaya untuk hari yang tersisa. Untuk informasi selengkapnya, lihat [Harga dan penagihan snapshot yang diarsipkan di Panduan Pengguna Amazon EBS](#).

Untuk menggunakan snapshot yang diarsipkan yang berada di Keranjang Sampah, Anda harus terlebih dahulu mengembalikan snapshot dari Keranjang Sampah dan kemudian memulihkannya dari tingkat arsip ke tingkat standar.

Kuota

Kuota berikut berlaku untuk Keranjang Sampah.

Kuota	Kuota default			
Aturan penyimpanan per Wilayah	250			
Pasangan kunci dan nilai tanda per aturan retensi	50			

Layanan-layanan terkait

Keranjang Sampah bekerja dengan layanan berikut:

- AWS CloudTrail — Memungkinkan Anda merekam peristiwa yang terjadi di Keranjang Sampah. Untuk informasi selengkapnya, lihat [Monitor Recycle Bin menggunakan AWS CloudTrail](#).

Harga

Sumber daya di Keranjang Sampah dikenai biaya dengan tarif standarnya. Tidak ada biaya tambahan untuk menggunakan Keranjang Sampah dan aturan penyimpanan. Untuk informasi selengkapnya, lihat [harga Amazon EBS](#).

Note

Beberapa sumber daya mungkin masih muncul di konsol Recycle Bin atau di output API AWS CLI dan untuk waktu yang singkat setelah periode retensi mereka kedaluwarsa dan telah dihapus secara permanen. Anda tidak dikenai biaya untuk sumber daya ini. Penagihan berhenti segera setelah periode retensi berakhir.

Anda dapat menggunakan tag alokasi biaya AWS yang dihasilkan berikut untuk tujuan pelacakan biaya dan alokasi saat menggunakan AWS Billing and Cost Management

- Kunci: `aws:recycle-bin:resource-in-bin`
- Nilai: `true`

Untuk informasi selengkapnya, lihat [Tanda alokasi biaya yang Dibuat AWS](#) di Panduan Pengguna AWS Billing and Cost Management .

Izin IAM yang diperlukan

Secara default, pengguna tidak memiliki izin untuk menggunakan Keranjang Sampah, aturan retensi, atau sumber daya yang ada di Keranjang Sampah. Untuk mengizinkan pengguna bekerja dengan sumber daya ini, Anda harus membuat kebijakan IAM yang memberikan izin menggunakan sumber daya dan tindakan API tertentu. Setelah kebijakan dibuat, Anda harus menambahkan izin ke pengguna, grup, atau peran Anda.

Topik

- [Izin untuk menggunakan Keranjang Sampah dan aturan retensi](#)
- [Izin untuk menggunakan sumber daya di Keranjang Sampah](#)

- [Kunci syarat untuk Keranjang Sampah](#)

Izin untuk menggunakan Keranjang Sampah dan aturan retensi

Untuk menggunakan Keranjang Sampah dan aturan retensi, pengguna memerlukan izin berikut.

- `rbin:CreateRule`
- `rbin:UpdateRule`
- `rbin:GetRule`
- `rbin:ListRules`
- `rbin>DeleteRule`
- `rbin:TagResource`
- `rbin:UntagResource`
- `rbin:ListTagsForResource`
- `rbin:LockRule`
- `rbin:UnlockRule`

Untuk menggunakan konsol Keranjang Sampah, pengguna memerlukan izin `tag:GetResources`.

Berikut ini adalah contoh kebijakan IAM yang menyertakan izin `tag:GetResources` untuk pengguna konsol. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "rbin:CreateRule",
      "rbin:UpdateRule",
      "rbin:GetRule",
      "rbin:ListRules",
      "rbin>DeleteRule",
      "rbin:TagResource",
      "rbin:UntagResource",
      "rbin:ListTagsForResource",
      "rbin:LockRule",
      "rbin:UnlockRule",
      "tag:GetResources"
    ]
  }]
}
```

```
    ],  
    "Resource": "*"    
  }]  
}
```

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
- (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Izin untuk menggunakan sumber daya di Keranjang Sampah

Untuk informasi selengkapnya tentang izin IAM yang dibutuhkan untuk menggunakan sumber daya di Keranjang Sampah, lihat aturan berikut ini:

- [Izin untuk bekerja dengan snapshot di Keranjang Sampah](#)
- [Izin untuk bekerja dengan AMI di Keranjang Sampah](#)

Kunci syarat untuk Keranjang Sampah

Keranjang Sampah menentukan kunci syarat berikut yang dapat Anda gunakan dalam elemen `Condition` dari kebijakan IAM untuk mengontrol kondisi di mana pernyataan kebijakan berlaku. Untuk informasi selengkapnya, lihat [Elemen kebijakan JSON IAM](#) di Panduan Pengguna IAM.

Topik

- [Kunci syarat `rbin:Request/ResourceType`](#)

- [Kunci syarat rbin:Attribute/ResourceType](#)

Kunci syarat **rbin:Request/ResourceType**

Kunci `rbin:Request/ResourceType` kondisi dapat digunakan untuk memfilter akses [CreateRule](#) dan [ListRules](#) permintaan berdasarkan nilai yang ditentukan untuk parameter `ResourceType` permintaan.

Contoh 1 - CreateRule

Contoh kebijakan IAM berikut memungkinkan prinsipal IAM untuk membuat `CreateRule` permintaan hanya jika nilai yang ditentukan untuk parameter permintaan adalah `ResourceType` atau `EBS_SNAPSHOT` `EC2_IMAGE`. Hal ini memungkinkan pengguna utama membuat aturan retensi baru hanya untuk snapshot dan AMI.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "rbin:CreateRule"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "rbin:Request/ResourceType" : ["EBS_SNAPSHOT", "EC2_IMAGE"]
        }
      }
    }
  ]
}
```

Contoh 2 - ListRules

Contoh kebijakan IAM berikut memungkinkan prinsipal IAM untuk membuat `ListRules` permintaan hanya jika nilai yang ditentukan untuk parameter permintaan adalah `ResourceType` `EBS_SNAPSHOT`. Hal ini memungkinkan pengguna utama membuat daftar aturan retensi hanya untuk snapshot, dan ini mencegahnya membuat aturan retensi untuk tipe sumber daya lainnya.

```
{
```

```

"Version" : "2012-10-17",
"Statement" : [
  {
    "Effect" : "Allow",
    "Action" : [
      "rbin:ListRules"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "rbin:Request/ResourceType" : "EBS_SNAPSHOT"
      }
    }
  }
]
}

```

Kunci syarat **rbin:Attribute/ResourceType**

Kunci `rbin:Attribute/ResourceType` kondisi dapat digunakan untuk memfilter akses pada [DeleteRuleGetRule](#), [UpdateRule](#), [LockRule](#), [UnlockRule](#), [TagResource](#), [UntagResource](#),,, dan [ListTagsForResource](#) permintaan berdasarkan nilai `ResourceType` atribut aturan retensi.

Contoh 1 - UpdateRule

Contoh kebijakan IAM berikut memungkinkan prinsipal IAM untuk membuat `UpdateRule` permintaan hanya jika `ResourceType` atribut dari aturan retensi yang diminta adalah atau. `EBS_SNAPSHOT` `EC2_IMAGE` Hal ini memungkinkan pengguna utama memperbarui aturan retensi hanya untuk snapshot dan AMI.

```

{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "rbin:UpdateRule"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "rbin:Attribute/ResourceType" : ["EBS_SNAPSHOT", "EC2_IMAGE"]
        }
      }
    }
  ]
}

```

```

    }
  }
]
}

```

Contoh 2 - DeleteRule

Contoh kebijakan IAM berikut memungkinkan prinsipal IAM untuk membuat DeleteRule permintaan hanya jika ResourceType atribut dari aturan retensi yang diminta adalah EBS_SNAPSHOT. Hal ini memungkinkan pengguna utama menghapus aturan retensi hanya untuk snapshot.

```

{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "rbin:DeleteRule"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "rbin:Attribute/ResourceType" : "EBS_SNAPSHOT"
        }
      }
    }
  ]
}

```

Bekerja dengan aturan retensi

Untuk mengaktifkan dan menggunakan Recycle Bin, Anda harus membuat aturan retensi di AWS Wilayah tempat Anda ingin melindungi sumber daya Anda. Aturan retensi menentukan hal berikut:

- Tipe sumber daya yang ingin Anda lindungi.
- Sumber daya yang ingin Anda pertahankan di Keranjang Sampah saat dihapus.
- Periode retensi untuk mempertahankan sumber daya di Keranjang Sampah sebelum dihapus secara permanen.

Dengan Keranjang Sampah, Anda dapat membuat dua tipe aturan retensi:

- **Aturan retensi tingkat tanda** — Aturan retensi tingkat tanda menggunakan tanda sumber daya untuk mengidentifikasi sumber daya yang akan dipertahankan di Keranjang Sampah. Untuk setiap aturan retensi, Anda menentukan satu atau beberapa pasangan kunci dan nilai tanda. Sumber daya tipe tertentu yang ditandai dengan setidaknya satu pasangan kunci dan nilai tanda yang ditentukan dalam aturan retensi secara otomatis dipertahankan di Keranjang Sampah setelah penghapusan. Gunakan tipe aturan retensi ini jika ingin melindungi sumber daya tertentu di akun Anda berdasarkan tandanya.
- **Aturan retensi tingkat wilayah** — Aturan retensi tingkat wilayah tidak memiliki tanda sumber daya yang ditentukan. Aturan ini berlaku untuk semua sumber daya dari tipe tertentu di Wilayah tempat aturan dibuat, bahkan jika sumber daya tidak ditandai. Gunakan tipe aturan retensi ini jika Anda ingin melindungi semua sumber daya tipe tertentu di Wilayah tertentu.

Setelah Anda membuat aturan retensi, sumber daya yang sesuai dengan kriterianya secara otomatis dipertahankan di Keranjang Sampah untuk periode retensi yang ditentukan setelah sumber daya tersebut dihapus.

Topik

- [Membuat aturan retensi](#)
- [Lihat aturan retensi Keranjang Sampah](#)
- [Untuk memperbarui aturan retensi](#)
- [Mengunci aturan retensi](#)
- [Buka kunci aturan retensi](#)
- [Menandai aturan retensi](#)
- [Melihat tanda aturan retensi](#)
- [Menghapus tanda dari aturan retensi](#)
- [Hapus aturan retensi Keranjang Sampah](#)

Membuat aturan retensi

Saat membuat aturan retensi, Anda harus menentukan parameter yang diperlukan berikut:

- Tipe sumber daya yang harus dilindungi oleh aturan retensi.
- Sumber daya yang harus dilindungi oleh aturan retensi. Anda dapat membuat aturan retensi pada tingkat tanda dan tingkat Wilayah.

- Untuk membuat aturan retensi tingkat tanda, tentukan tanda sumber daya yang mengidentifikasi sumber daya yang akan dilindungi. Anda dapat menentukan hingga 50 tanda untuk setiap aturan, dan menambahkan pasangan kunci dan nilai tanda yang sama ke maksimum lima aturan retensi.
- Untuk membuat aturan retensi tingkat Wilayah, jangan menentukan pasangan kunci dan nilai tanda apa pun. Dalam hal ini, semua sumber daya dari tipe tertentu dilindungi.
- Periode untuk mempertahankan sumber daya di Keranjang Sampah setelah dihapus. Periode bisa sampai 1 tahun (365 hari).

Anda juga dapat menentukan parameter opsional berikut:

- Nama opsional untuk aturan retensi. Nama dapat memuat hingga 255 karakter.
- Nama opsional untuk deskripsi retensi. Deskripsi dapat memuat hingga 255 karakter.

 Note

Kami menyarankan Anda untuk tidak memasukkan informasi identitas pribadi, rahasia, atau sensitif dalam deskripsi aturan retensi.

- Tanda aturan retensi opsional untuk membantu mengidentifikasi dan mengatur aturan retensi Anda. Anda dapat menetapkan hingga 50 tanda untuk setiap aturan.

Anda juga dapat secara opsional mengunci aturan retensi pada saat pembuatan. Jika mengunci aturan retensi pada pembuatan, Anda juga harus menentukan periode penundaan pembukaan kunci, 7 hingga 30 hari. Aturan retensi tetap tidak terkunci secara default kecuali Anda menguncinya secara eksplisit.

Aturan retensi hanya berfungsi di Wilayah tempat pembuatannya. Jika ingin menggunakan Keranjang Sampah di Wilayah lain, Anda harus membuat aturan retensi tambahan di Wilayah tersebut.

Anda dapat membuat sebuah aturan retensi Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk membuat aturan retensi

1. [Buka konsol Keranjang Sampah di https://console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)

2. Di panel navigasi, pilih buat Aturan retensi, lalu pilih Buat aturan retensi.
3. Di bagian Detail aturan, lakukan hal berikut:
 - a. (Opsional) Untuk Nama aturan retensi, masukkan nama deskriptif untuk aturan retensi.
 - b. (Opsional) Untuk Deskripsi aturan retensi, masukkan deskripsi singkat untuk aturan retensi.
4. Di bagian Pengaturan aturan, lakukan hal berikut ini:
 - a. Untuk Tipe sumber daya, pilih tipe sumber daya yang akan dilindungi oleh aturan retensi. Aturan retensi hanya akan mempertahankan sumber daya dari tipe ini di Keranjang Sampah.
 - b. Lakukan salah satu langkah berikut:
 - Untuk membuat aturan retensi tingkat Wilayah yang cocok dengan semua sumber daya yang dihapus dari tipe tertentu di Wilayah, pilih Terapkan ke semua sumber daya. Aturan retensi akan mempertahankan semua sumber daya yang dihapus dari yang ditentukan di dalam Keranjang Sampah setelah penghapusan, bahkan jika sumber daya tidak memiliki tanda apa pun.
 - Guna membuat aturan retensi tingkat tanda, untuk Tanda sumber daya yang akan dicocokkan, masukkan pasangan kunci dan nilai tanda yang akan digunakan untuk mengidentifikasi sumber daya dari tipe tertentu yang akan dipertahankan di Keranjang Sampah. Hanya sumber daya dari tipe tertentu yang memiliki setidaknya satu dari pasangan kunci dan nilai tanda tertentu yang akan dipertahankan oleh aturan retensi.
 - c. Untuk Periode retensi, masukkan jumlah hari aturan retensi untuk mempertahankan sumber daya di Keranjang Sampah.
5. (Opsional) Guna mengunci aturan retensi, untuk Pengaturan penguncian aturan, pilih Kunci, lalu untuk Membuka kunci periode penundaan, tentukan periode penundaan pembukaan kunci dalam hari. Aturan retensi terkunci tidak dapat diubah atau dihapus. Untuk mengubah atau menghapus aturan, Anda harus terlebih dahulu membukanya dan kemudian menunggu periode penundaan pembukaan kunci berakhir. Untuk informasi selengkapnya, lihat [Mengunci aturan retensi](#)

Agar aturan retensi tidak terkunci, untuk Pengaturan penguncian aturan, tetap pilih Buka kunci. Aturan retensi yang tidak terkunci dapat diubah atau dihapus kapan saja. Untuk informasi selengkapnya, lihat [Buka kunci aturan retensi](#).

6. (Opsional) Di bagian Tanda, lakukan hal berikut:

- Untuk menandai aturan dengan tanda kustom, pilih Tambahkan tanda, lalu masukkan pasangan kunci dan nilai tanda.

7. Pilih Buat aturan retensi.

AWS CLI

Untuk membuat aturan retensi

Gunakan perintah AWS CLI [create-rule](#). Untuk `--retention-period`, tentukan jumlah hari guna mempertahankan snapshot yang terhapus di Keranjang Sampah. Untuk `--resource-type`, tentukan `EBS_SNAPSHOT` untuk snapshot atau `EC2_IMAGE` untuk AMI. Untuk membuat aturan retensi tingkat tanda, untuk `--resource-tags`, tentukan tanda yang akan digunakan guna mengidentifikasi snapshot yang akan dipertahankan. Untuk membuat aturan retensi tingkat Wilayah, hilangkan `--resource-tags`. Untuk mengunci aturan retensi, sertakan `--lock-configuration`, dan tentukan periode penundaan pembukaan kunci dalam hari.

```
aws rbin create-rule \  
--retention-period RetentionPeriodValue=number_of_days,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT|EC2_IMAGE \  
--description "rule_description" \  
--lock-configuration  
'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=unlock_delay_in_days}' \  
--resource-tags ResourceTagKey=tag_key,ResourceTagValue=tag_value
```

Contoh 1

Perintah contoh berikut membuat aturan retensi tingkat Wilayah yang tidak terkunci yang mempertahankan semua snapshot yang dihapus selama 7 hari.

```
aws rbin create-rule \  
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT \  
--description "Match all snapshots"
```

Contoh 2

Contoh perintah berikut membuat aturan tingkat tanda yang mempertahankan snapshot yang dihapus yang ditandai dengan `purpose=production` untuk jangka waktu 7 hari.

```
aws rbin create-rule \  
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT \  
--description "Match snapshots with a specific tag" \  
--resource-tags ResourceTagKey=purpose,ResourceTagValue=production
```

Contoh 3

Contoh perintah berikut membuat aturan retensi tingkat Wilayah terkunci yang mempertahankan semua snapshot yang dihapus selama jangka waktu 7 hari. Aturan retensi dikunci dengan periode penundaan buka kunci 7 hari.

```
aws rbin create-rule \  
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT \  
--description "Match all snapshots" \  
--lock-configuration 'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=7}'
```

Lihat aturan retensi Keranjang Sampah

Anda dapat melihat aturan retensi Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk melihat aturan retensi

1. [Buka konsol Keranjang Sampah di https://console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)
2. Di panel navigasi, pilih Aturan retensi.
3. Grid mencantumkan semua aturan retensi untuk Wilayah yang dipilih. Untuk melihat lebih banyak informasi tentang aturan retensi tertentu, pilih aturan penyimpanan tertentu di kisi.

AWS CLI

Untuk melihat semua aturan retensi Anda

Gunakan perintah AWS CLI [list-rules](#), dan untuk `--resource-type` tentukan `EBS_SNAPSHOT` untuk snapshot atau `EC2_IMAGE` untuk AMI.

```
aws rbin list-rules --resource-type EBS_SNAPSHOT|EC2_IMAGE
```

Contoh

Perintah contoh berikut menyediakan daftar semua aturan retensi yang mempertahankan snapshot.

```
aws rbin list-rules --resource-type EBS_SNAPSHOT
```

Untuk melihat informasi untuk aturan retensi tertentu

Gunakan perintah [get-rule](#) AWS CLI .

```
aws rbin get-rule --identifier rule_ID
```

Contoh

Contoh perintah berikut menyediakan informasi tentang pwxIkFcvge4 aturan retensi.

```
aws rbin get-rule --identifier pwxIkFcvge4
```

Untuk memperbarui aturan retensi

Anda dapat memperbarui deskripsi aturan retensi yang tidak terkunci, tanda sumber daya, dan periode retensi kapan saja setelah pembuatan. Anda tidak dapat memperbarui tipe sumber daya aturan retensi atau membuka periode penundaan, meskipun aturan retensi tidak terkunci.

Anda tidak dapat memperbarui aturan penyimpanan terkunci dengan cara apa pun. Jika perlu mengubah aturan retensi yang terkunci, Anda harus terlebih dahulu membukanya dan menunggu periode penundaan pembukaan kunci berakhir.

Jika perlu mengubah periode penundaan pembukaan kunci untuk aturan retensi yang terkunci, Anda harus [membuka aturan retensi](#), dan menunggu periode penundaan pembukaan kunci saat ini berakhir. Ketika periode penundaan pembukaan kunci berakhir, Anda harus [mengunci kembali aturan retensi](#) dan menentukan periode penundaan pembukaan kunci yang baru.

Note

Kami menyarankan Anda untuk tidak memasukkan informasi identitas pribadi, rahasia, atau sensitif dalam deskripsi aturan retensi.

Setelah Anda memperbarui aturan retensi, perubahan hanya berlaku untuk sumber daya baru yang dipertahankan. Perubahan tidak memengaruhi sumber daya yang sebelumnya dikirim ke Keranjang Sampah. Misalnya, jika Anda memperbarui periode retensi aturan retensi, hanya snapshot yang dihapus setelah pembaruan dipertahankan untuk periode retensi baru. Snapshot yang dikirim ke Keranjang Sampah sebelum pembaruan masih akan dipertahankan untuk periode retensi (lama) sebelumnya.

Anda dapat memperbarui aturan retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk memperbarui aturan retensi

1. [Buka konsol Keranjang Sampah di https://console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)
2. Di panel navigasi, pilih Aturan retensi.
3. Di grid, pilih aturan retensi yang akan dihapus, dan pilih Tindakan, Hapus aturan retensi.
4. Di bagian Detail aturan, perbarui Nama aturan retensi dan Deskripsi aturan retensi sesuai kebutuhan.
5. Di bagian Pengaturan aturan, perbarui Tipe sumber daya, Tanda sumber daya yang akan dicocokkan, dan Periode retensi sesuai kebutuhan.
6. Di bagian Tanda, tambahkan atau hapus tanda aturan retensi sesuai kebutuhan.
7. Pilih Simpan aturan retensi.

AWS CLI

Untuk memperbarui aturan retensi

Gunakan perintah AWS CLI [update-rule](#). Untuk `--identifier`, tentukan ID aturan retensi yang akan diperbarui. Untuk `--resource-types`, tentukan `EBS_SNAPSHOT` untuk snapshot atau `EC2_IMAGE` untuk AMI.

```
aws rbin update-rule \  
--identifier rule_ID \  
--retention-period RetentionPeriodValue=number_of_days,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT|EC2_IMAGE \  
--description "rule_description"
```

Contoh

Perintah contoh berikut memperbarui aturan retensi 61sJ2Fa9nh9 untuk mempertahankan semua snapshot selama 7 hari dan memperbarui deskripsinya.

```
aws rbin update-rule \  
--identifier 61sJ2Fa9nh9 \  
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \  
--resource-type EBS_SNAPSHOT \  
--description "Retain for three weeks"
```

Mengunci aturan retensi

Keranjang Sampah memungkinkan Anda mengunci aturan retensi tingkat Wilayah kapan saja.

Note

Anda tidak dapat mengunci aturan retensi tingkat tanda.

Aturan retensi yang terkunci tidak dapat dimodifikasi atau dihapus, bahkan oleh pengguna yang memiliki izin IAM yang diperlukan. Kunci aturan retensi Anda untuk membantu melindunginya dari modifikasi dan penghapusan yang tidak disengaja atau berbahaya.

Saat mengunci aturan retensi, Anda harus menentukan periode penundaan pembukaan kunci. Periode ini adalah periode waktu yang harus Anda tunggu setelah membuka kunci aturan retensi sebelum dapat memodifikasi atau menghapusnya. Anda tidak dapat memodifikasi atau menghapus aturan retensi selama periode penundaan pembukaan kunci. Anda dapat memodifikasi atau menghapus aturan retensi hanya setelah periode penundaan pembukaan kunci berakhir.

Anda tidak dapat mengubah periode penundaan pembukaan kunci setelah aturan retensi terkunci. Jika izin akun Anda telah disusupi, periode penundaan pembukaan kunci memberi Anda waktu tambahan untuk mendeteksi dan merespons ancaman keamanan. Jangka waktu periode ini harus lebih lama dari waktu yang Anda butuhkan untuk mengidentifikasi dan merespons pelanggaran keamanan. Untuk menetapkan durasi yang tepat, Anda dapat meninjau insiden keamanan sebelumnya dan waktu yang diperlukan untuk mengidentifikasi serta meremediasi pelanggaran akun.

Sebaiknya gunakan EventBridge aturan Amazon untuk memberi tahu Anda tentang perubahan status kunci aturan retensi. Untuk informasi selengkapnya, lihat [Pantau Recycle Bin menggunakan Amazon EventBridge](#).

Pertimbangan

- Anda hanya dapat mengunci aturan retensi tingkat Wilayah.
- Anda dapat mengunci aturan retensi yang tidak terkunci kapan saja.
- Periode penundaan pembukaan kunci harus selama 7 hingga 30 hari.
- Anda dapat mengunci kembali aturan retensi selama periode penundaan pembukaan kunci. Mengunci kembali aturan retensi akan mereset periode penundaan pembukaan kunci.

Anda dapat mengunci aturan retensi tingkat Wilayah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk mengunci aturan retensi

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Aturan retensi.
3. Di kisi, pilih aturan retensi yang tidak terkunci untuk dikunci, dan pilih Tindakan, Edit kunci aturan retensi.
4. Di layar Edit aturan retensi, pilih Kunci, lalu untuk Buka kunci periode penundaan, tentukan periode penundaan pembukaan kunci dalam beberapa hari.
5. Pilih kotak centang Saya memahami bahwa mengunci aturan retensi akan mencegahnya dari modifikasi atau penghapusan, lalu pilih Simpan.

AWS CLI

Untuk mengunci aturan retensi yang tidak terkunci

Gunakan perintah AWS CLI [lock-rule](#). Untuk `--identifier`, tentukan ID dari aturan retensi yang akan dikunci. Untuk `--lock-configuration`, tentukan periode penundaan pembukaan kunci dalam beberapa hari.

```
aws rbin lock-rule \  
--identifier rule_ID \  
--lock-configuration  
'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=number_of_days}'
```

Contoh

Perintah contoh berikut mengunci aturan retensi 61sJ2Fa9nh9 dan menetapkan periode penundaan pembukaan kunci menjadi 15 hari.

```
aws rbin lock-rule \  
--identifier 61sJ2Fa9nh9 \  
--lock-configuration 'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=15}'
```

Buka kunci aturan retensi

Anda tidak dapat memodifikasi atau menghapus aturan retensi yang terkunci. Jika perlu memodifikasi aturan retensi yang terkunci, Anda harus membukanya terlebih dahulu. Setelah Anda membuka kunci aturan retensi, Anda harus menunggu periode penundaan pembukaan kunci berakhir sebelum Anda memodifikasi atau menghapusnya. Anda tidak dapat memodifikasi atau menghapus aturan retensi selama periode penundaan pembukaan kunci.

Aturan retensi yang tidak terkunci dapat dimodifikasi dan dihapus kapan saja oleh pengguna yang memiliki izin IAM yang diperlukan. Membiarkan aturan retensi tidak terkunci dapat mengeksposnya dari modifikasi dan penghapusan yang tidak disengaja atau berbahaya.

Pertimbangan

- Anda dapat mengunci kembali aturan retensi selama periode penundaan pembukaan kunci.
- Anda dapat mengunci kembali aturan retensi setelah periode penundaan pembukaan kunci berakhir.
- Anda tidak dapat melewati periode penundaan pembukaan kunci.
- Anda tidak dapat mengubah periode penundaan pembukaan kunci setelah penguncian awal.

Sebaiknya gunakan EventBridge aturan Amazon untuk memberi tahu Anda tentang perubahan status kunci aturan retensi. Untuk informasi selengkapnya, lihat [Pantau Recycle Bin menggunakan Amazon EventBridge](#).

Anda dapat membuka kunci aturan retensi tingkat Wilayah yang terkunci menggunakan salah satu metode berikut.

Recycle Bin console

Untuk membuka kunci aturan retensi

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Aturan retensi.
3. Di grid, pilih aturan retensi terkunci yang akan dibuka, dan pilih Tindakan, Edit kunci aturan retensi.
4. Pada layar Edit kunci aturan retensi, pilih Buka kunci, lalu pilih Simpan.

AWS CLI

Untuk membuka aturan retensi yang terkunci

Gunakan perintah AWS CLI [lock-rule](#). Untuk `--identifier`, tentukan ID dari aturan retensi yang akan dikunci.

```
aws rbin unlock-rule \  
--identifier rule_ID
```

Contoh

Perintah contoh berikut membuka aturan retensi 61sJ2Fa9nh9

```
aws rbin unlock-rule \  
--identifier 61sJ2Fa9nh9
```

Menandai aturan retensi

Anda dapat menetapkan tanda kustom ke aturan penyimpanan untuk mengkategorikannya dengan cara berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Hal ini membantu menemukan aturan retensi tertentu berdasarkan tanda kustom yang Anda tetapkan.

Anda dapat menetapkan tanda ke sebuah aturan retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk menandai aturan retensi

1. [Buka konsol Keranjang Sampah di https://console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)
2. Di panel navigasi, pilih Aturan retensi.
3. Pilih aturan retensi yang akan diberi tanda, pilih tab Tanda, lalu pilih Kelola tanda.
4. Pilih Tambahkan tanda. Untuk Kunci, masukkan kunci tanda. Untuk Nilai, masukkan nilai tanda.
5. Pilih Simpan.

AWS CLI

Untuk menandai aturan retensi

Gunakan perintah [tag-resource](#) AWS CLI . Untuk `--resource-arn`, tentukan Amazon Resource Name (ARN) dari aturan retensi yang akan ditandai, dan untuk `--tags`, tentukan kunci tanda dan pasangan nilainya.

```
aws rbin tag-resource \  
--resource-arn retention_rule_arn \  
--tags key=tag_key,value=tag_value
```

Contoh

Berikut contoh perintah tanda aturan retensi `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3` dengan tanda `purpose=production`.

```
aws rbin tag-resource \  
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3 \  
--tags key=purpose,value=production
```

Melihat tanda aturan retensi

Anda dapat melihat tanda yang ditetapkan untuk aturan retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk melihat tanda aturan retensi

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Aturan retensi.
3. Pilih aturan retensi untuk melihat tanda, lalu pilih tab Tanda.

AWS CLI

Untuk melihat tanda yang ditetapkan ke aturan retensi

Gunakan perintah AWS CLI [list-tags-for-resource](#). Untuk `--resource-arn`, tentukan ARN dari aturan retensi.

```
aws rbin list-tags-for-resource \  
--resource-arn retention_rule_arn
```

Contoh

Contoh perintah berikut mencantumkan tanda untuk aturan retensi `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3`.

```
aws rbin list-tags-for-resource \  
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3
```

Menghapus tanda dari aturan retensi

Anda dapat menghapus tanda dari sebuah aturan retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk menghapus tanda dari aturan retensi

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Aturan retensi.
3. Pilih aturan retensi untuk menghapus tanda, pilih tab Tanda, lalu pilih Kelola tanda.
4. Pilih Hapus di sebelah tanda yang akan dihapus.

5. Pilih Simpan.

AWS CLI

Untuk menghapus tanda dari aturan retensi

Gunakan perintah [untag-resource](#) AWS CLI . Untuk `--resource-arn`, tentukan ARN dari aturan retensi. Untuk `--tagkeys`, tentukan kunci tanda dari tanda yang akan dihapus.

```
aws rbin untag-resource \  
--resource-arn retention_rule_arn \  
--tagkeys tag_key
```

Contoh

Contoh perintah berikut menghapus tanda yang memiliki kunci tanda `purpose` dari aturan retensi `arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3`.

```
aws rbin untag-resource \  
--resource-arn arn:aws:rbin:us-east-1:123456789012:rule/n0oSBBtItF3 \  
--tagkeys purpose
```

Hapus aturan retensi Keranjang Sampah

Anda dapat menghapus aturan retensi kapan saja. Saat Anda menghapus aturan retensi, aturan tersebut tidak lagi mempertahankan sumber daya baru di Keranjang Sampah setelah dihapus. Sumber daya yang dikirim ke Keranjang Sampah sebelum aturan retensi dihapus terus disimpan di Keranjang Sampah sesuai dengan periode retensi yang ditentukan dalam aturan retensi. Ketika periode berakhir, sumber daya dihapus secara permanen dari Keranjang Sampah.

Anda dapat menghapus aturan retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk menghapus aturan retensi

1. [Buka konsol Keranjang Sampah di https://console.aws.amazon.com/rbin/home/](https://console.aws.amazon.com/rbin/home/)
2. Di panel navigasi, pilih Aturan retensi.
3. Di grid, pilih aturan retensi yang akan dihapus, dan pilih Tindakan, Hapus aturan retensi.

4. Saat diminta, masukkan pesan konfirmasi dan pilih Hapus aturan retensi.

AWS CLI

Untuk menghapus aturan retensi

Gunakan perintah AWS CLI [delete-rule](#). Untuk `--identifier`, tentukan ID dari aturan retensi yang akan dihapus.

```
aws rbin delete-rule --identifier rule_ID
```

Contoh

Perintah contoh berikut menghapus aturan retensi 61sJ2Fa9nh9.

```
aws rbin delete-rule --identifier 61sJ2Fa9nh9
```

Bekerja dengan sumber daya di Keranjang Sampah

Keranjang Sampah mendukung tipe sumber daya berikut:

- Snapshot Amazon EBS
- Amazon Machine Images (AMI) yang didukung Amazon EBS

Tugas

- [Memulihkan snapshot dari Keranjang Sampah](#)
- [Memulihkan AMI dari Keranjang Sampah](#)

Memulihkan snapshot dari Keranjang Sampah

Keranjang Sampah adalah fitur pemulihan data yang memungkinkan Anda memulihkan snapshot Amazon EBS dan AMI yang didukung EBS yang terhapus secara tidak sengaja. Saat menggunakan Keranjang Sampah, jika sumber daya Anda dihapus, sumber daya tersebut dipertahankan di Keranjang Sampah untuk jangka waktu yang Anda tentukan sebelum dihapus secara permanen.

Anda dapat memulihkan sumber daya dari Keranjang Sampah kapan saja sebelum periode retensi berakhir. Setelah memulihkan sumber daya dari Keranjang Sampah, sumber daya tersebut akan

dihapus dari Keranjang Sampah dan Anda dapat menggunakannya dengan cara yang sama seperti menggunakan sumber daya lain dari tipe tersebut di akun Anda. Jika periode retensi berakhir dan sumber daya tidak dipulihkan, sumber daya tersebut akan dihapus secara permanen dari Keranjang Sampah dan tidak lagi tersedia untuk pemulihan.

Snapshot di Keranjang Sampah ditagih dengan tarif yang sama dengan snapshot reguler di akun Anda. Tidak ada biaya tambahan untuk menggunakan Keranjang Sampah dan aturan penyimpanan. Untuk informasi selengkapnya, lihat [harga Amazon EBS](#).

Untuk informasi selengkapnya, lihat [Keranjang Sampah](#).

Topik

- [Izin untuk bekerja dengan snapshot di Keranjang Sampah](#)
- [Lihat snapshot di Keranjang Sampah](#)
- [Mengembalikan snapshot dari Keranjang Sampah](#)

Izin untuk bekerja dengan snapshot di Keranjang Sampah

Secara default, pengguna tidak memiliki izin untuk bekerja dengan snapshot yang ada di Keranjang Sampah. Untuk mengizinkan pengguna bekerja dengan sumber daya ini, Anda harus membuat kebijakan IAM yang memberikan izin menggunakan sumber daya dan tindakan API tertentu. Setelah kebijakan dibuat, Anda harus menambahkan izin ke pengguna, grup, atau peran Anda.

Untuk melihat dan memulihkan snapshot yang ada di Keranjang Sampah, pengguna harus memiliki izin berikut:

- `ec2:ListSnapshotsInRecycleBin`
- `ec2:RestoreSnapshotFromRecycleBin`

Untuk mengelola tanda untuk snapshot di Keranjang Sampah, pengguna memerlukan izin tambahan berikut.

- `ec2:CreateTags`
- `ec2:DeleteTags`

Untuk menggunakan konsol Keranjang Sampah, pengguna memerlukan `ec2:DescribeTags` izin.

Berikut ini adalah contoh kebijakan IAM. Ini termasuk izin `ec2:DescribeTags` untuk pengguna konsol, dan itu termasuk izin `ec2:CreateTags` dan `ec2>DeleteTags` untuk mengelola tag. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:ListSnapshotsInRecycleBin",
        "ec2:RestoreSnapshotFromRecycleBin"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2>DeleteTags",
        "ec2:DescribeTags"
      ],
      "Resource": "arn:aws:ec2:Region:account-id:snapshot/*"
    }
  ]
}
```

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.

- (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang izin yang diperlukan untuk menggunakan Keranjang Sampah, lihat [Izin untuk menggunakan Keranjang Sampah dan aturan retensi](#).

Lihat snapshot di Keranjang Sampah

Saat snapshot ada di Keranjang Sampah, Anda dapat melihat informasi terbatas tentangnya, termasuk:

- ID snapshot.
- Deskripsi snapshot.
- ID volume tempat snapshot dibuat.
- Tanggal dan waktu snapshot dihapus dan masuk Keranjang Sampah.
- Tanggal dan waktu ketika periode retensi kedaluwarsa. Snapshot akan dihapus secara permanen dari Keranjang Sampah saat ini.

Anda dapat melihat snapshot di Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk melihat snapshot di Keranjang Sampah menggunakan konsol

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Keranjang Sampah.
3. Kisi mencantumkan semua snapshot yang saat ini ada di Keranjang Sampah. Untuk melihat detail AMI tertentu, pilih di kisi, dan pilih Tindakan, Lihat detail.

AWS CLI

Untuk melihat snapshot di Recycle Bin menggunakan AWS CLI

Gunakan perintah [list-snapshots-in-recycle-bin](#) AWS CLI . Sertakan opsi `--snapshot-id` untuk melihat snapshot tertentu. Atau hilangkan opsi `--snapshot-id` untuk melihat semua snapshot di Keranjang Sampah.

```
aws ec2 list-snapshots-in-recycle-bin --snapshot-id snapshot_id
```

Misalnya, perintah berikut memberikan informasi tentang snapshot `snap-01234567890abcdef` di Keranjang Sampah.

```
aws ec2 list-snapshots-in-recycle-bin --snapshot-id snap-01234567890abcdef
```

Contoh output:

```
{
  "SnapshotRecycleBinInfo": [
    {
      "Description": "Monthly data backup snapshot",
      "RecycleBinEnterTime": "2021-12-01T13:00:00.000Z",
      "RecycleBinExitTime": "2021-12-15T13:00:00.000Z",
      "VolumeId": "vol-abcdef09876543210",
      "SnapshotId": "snap-01234567890abcdef"
    }
  ]
}
```

Mengembalikan snapshot dari Keranjang Sampah

Anda tidak dapat menggunakan snapshot dengan cara apa pun saat berada di Keranjang Sampah. Untuk menggunakan AMI, Anda harus memulihkannya terlebih dahulu. Saat Anda memulihkan snapshot dari Keranjang Sampah, snapshot segera tersedia untuk digunakan, dan akan dihapus dari Keranjang Sampah. Anda dapat menggunakan AMI yang dipulihkan dengan cara yang sama seperti Anda menggunakan AMI lainnya di akun Anda.

Anda dapat memulihkan snapshot dari Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk memulihkan snapshot dari Keranjang Sampah menggunakan konsol

1. Buka konsol Keranjang Sampah di <https://console.aws.amazon.com/rbin/home/>
2. Di panel navigasi, pilih Keranjang Sampah.
3. Kisi mencantumkan semua snapshot yang saat ini ada di Keranjang Sampah. Pilih snapshot yang akan dipulihkan, lalu pilih Pulihkan.

4. Saat diminta, pilih Pulihkan.

AWS CLI

Untuk mengembalikan snapshot yang dihapus dari Recycle Bin menggunakan AWS CLI

Gunakan perintah [AWS CLI restore-snapshot-from-recycle-bin](#). Untuk `--snapshot-id`, tentukan ID snapshot yang akan dipulihkan.

```
aws ec2 restore-snapshot-from-recycle-bin --snapshot-id snapshot_id
```

Misalnya, perintah berikut memulihkan snapshot `snap-01234567890abcdef` dari Keranjang Sampah.

```
aws ec2 restore-snapshot-from-recycle-bin --snapshot-id snap-01234567890abcdef
```

Contoh output:

```
{
  "SnapshotId": "snap-01234567890abcdef",
  "Description": "Monthly data backup snapshot",
  "Encrypted": false,
  "OwnerId": "111122223333",
  "Progress": "100%",
  "StartTime": "2021-12-01T13:00:00.000000+00:00",
  "State": "recovering",
  "VolumeId": "vol-ffffffff",
  "VolumeSize": 30
}
```

Memulihkan AMI dari Keranjang Sampah

Keranjang Sampah adalah fitur pemulihan data yang memungkinkan Anda memulihkan snapshot dan AMI yang didukung Amazon EBS yang terhapus secara tidak sengaja. Saat menggunakan Keranjang Sampah, jika sumber daya Anda dihapus, sumber daya tersebut dipertahankan di Keranjang Sampah untuk jangka waktu yang Anda tentukan sebelum dihapus secara permanen.

Anda dapat memulihkan sumber daya dari Keranjang Sampah kapan saja sebelum periode retensi berakhir. Setelah memulihkan sumber daya dari Keranjang Sampah, sumber daya tersebut akan

dihapus dari Keranjang Sampah dan Anda dapat menggunakannya dengan cara yang sama seperti menggunakan sumber daya lain dari tipe tersebut di akun Anda. Jika periode retensi berakhir dan sumber daya tidak dipulihkan, sumber daya tersebut dihapus secara permanen dari Keranjang Sampah dan tidak lagi tersedia untuk pemulihan.

AMI di Keranjang Sampah tidak dikenai biaya tambahan.

Untuk informasi selengkapnya, lihat [Keranjang Sampah](#).

Topik

- [Izin untuk bekerja dengan AMI di Keranjang Sampah](#)
- [Melihat AMI di Keranjang Sampah](#)
- [Memulihkan AMI dari Keranjang Sampah](#)

Izin untuk bekerja dengan AMI di Keranjang Sampah

Secara default, pengguna tidak memiliki izin untuk bekerja dengan AMI yang ada di Keranjang Sampah. Untuk mengizinkan pengguna bekerja dengan sumber daya ini, Anda harus membuat kebijakan IAM yang memberikan izin menggunakan sumber daya dan tindakan API tertentu. Setelah kebijakan dibuat, Anda harus menambahkan izin ke pengguna, grup, atau peran Anda.

Untuk melihat dan memulihkan AMI yang ada di Keranjang Sampah, pengguna harus memiliki izin berikut:

- `ec2:ListImagesInRecycleBin`
- `ec2:RestoreImageFromRecycleBin`

Untuk mengelola tag untuk AMI di Keranjang Sampah, pengguna memerlukan izin tambahan berikut.

- `ec2:CreateTags`
- `ec2>DeleteTags`

Untuk menggunakan konsol Keranjang Sampah, pengguna memerlukan izin `ec2:DescribeTags`.

Berikut ini adalah contoh kebijakan IAM. Ini termasuk izin `ec2:DescribeTags` untuk pengguna konsol, dan itu termasuk izin `ec2:CreateTags` dan `ec2>DeleteTags` untuk mengelola tag. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:ListImagesInRecycleBin",
        "ec2:RestoreImageFromRecycleBin"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2>DeleteTags",
        "ec2:DescribeTags"
      ],
      "Resource": "arn:aws:ec2:Region::image/*"
    }
  ]
}
```

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Membuat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
- (Tidak disarankan) Pasang kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam [Menambahkan izin ke pengguna \(konsol\)](#) di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang izin yang diperlukan untuk menggunakan Keranjang Sampah, lihat [Izin untuk menggunakan Keranjang Sampah dan aturan retensi](#).

Melihat AMI di Keranjang Sampah

Saat AMI berada di Keranjang Sampah, Anda dapat melihat informasi terbatas tentangnya, termasuk:

- Nama, deskripsi, dan ID unik AMI.
- Tanggal dan waktu ketika AMI dihapus dan masuk Keranjang Sampah.
- Tanggal dan waktu ketika periode retensi kedaluwarsa. AMI akan dihapus secara permanen di waktu tersebut.

Anda dapat melihat AMI di Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk melihat AMI di Keranjang Sampah menggunakan konsol

1. Buka konsol Keranjang Sampah di console.aws.amazon.com/rbin/home/.
2. Di panel navigasi, pilih Keranjang Sampah.
3. Grid mencantumkan semua sumber daya yang saat ini ada di Keranjang Sampah. Untuk melihat detail untuk AMI tertentu, pilih di grid, dan pilih Tindakan, Lihat detail.

AWS CLI

Untuk melihat AMI yang dihapus di Recycle Bin menggunakan AWS CLI

Gunakan AWS CLI perintah [list-images-in-recycle-bin](#). Untuk melihat AMI tertentu, sertakan opsi `--image-id` dan tentukan ID AMI yang akan dilihat. Anda dapat menentukan hingga 20 ID dalam satu permintaan.

Untuk melihat semua AMI di Keranjang Sampah, hilangkan opsi `--image-id`. Jika Anda tidak menentukan nilai untuk `--max-items`, perintah mengembalikan 1.000 item per halaman, secara default. Untuk informasi selengkapnya, lihat [Paginasi](#) di Referensi API Amazon EC2.

```
aws ec2 list-images-in-recycle-bin --image-id ami_id
```

Misalnya, perintah berikut ini memberikan informasi tentang `ami-01234567890abcdef` AMI di Keranjang Sampah.

```
aws ec2 list-images-in-recycle-bin --image-id ami-01234567890abcdef
```

Contoh output:

```
{
  "Images": [
    {
      "ImageId": "ami-0f740206c743d75df",
      "Name": "My AL2 AMI",
      "Description": "My Amazon Linux 2 AMI",
      "RecycleBinEnterTime": "2021-11-26T21:04:50+00:00",
      "RecycleBinExitTime": "2022-03-06T21:04:50+00:00"
    }
  ]
}
```

Important

Jika Anda menerima kesalahan berikut, Anda mungkin perlu memperbarui AWS CLI versi Anda. Untuk informasi selengkapnya, lihat [Kesalahan perintah tidak ditemukan](#).

```
aws.exe: error: argument operation: Invalid choice, valid choices are: ...
```

Memulihkan AMI dari Keranjang Sampah

Anda tidak dapat menggunakan AMI dengan cara apa pun saat berada di Keranjang Sampah. Untuk menggunakan AMI, Anda harus memulihkannya terlebih dahulu. Saat Anda memulihkan AMI dari Keranjang Sampah, AMI segera tersedia untuk digunakan, dan akan dihapus dari Keranjang Sampah. Anda dapat menggunakan AMI yang dipulihkan dengan cara yang sama seperti Anda menggunakan AMI lainnya di akun Anda.

Anda dapat memulihkan AMI dari Keranjang Sampah menggunakan salah satu metode berikut.

Recycle Bin console

Untuk memulihkan AMI dari Keranjang Sampah menggunakan konsol

1. Buka konsol Keranjang Sampah di console.aws.amazon.com/rbin/home/.

2. Di panel navigasi, pilih Keranjang Sampah.
3. Grid mencantumkan semua sumber daya yang saat ini ada di Keranjang Sampah. Pilih AMI yang akan dipulihkan, lalu pilih Pulihkan.
4. Saat diminta, pilih Pulihkan.

AWS CLI

Untuk mengembalikan AMI yang dihapus dari Recycle Bin menggunakan AWS CLI

Gunakan AWS CLI perintah [restore-image-from-recycle-bin](#). Untuk `--image-id`, tentukan ID AMI yang akan dipulihkan.

```
aws ec2 restore-image-from-recycle-bin --image-id ami_id
```

Misalnya, perintah berikut ini memulihkan `ami-01234567890abcdef` AMI dari Keranjang Sampah.

```
aws ec2 restore-image-from-recycle-bin --image-id ami-01234567890abcdef
```

Jika berhasil, perintah ini tidak memunculkan output.

Important

Jika Anda menerima kesalahan berikut, Anda mungkin perlu memperbarui AWS CLI versi Anda. Untuk informasi selengkapnya, lihat [Kesalahan perintah tidak ditemukan](#).

```
aws.exe: error: argument operation: Invalid choice, valid choices are: ...
```

Pantau Keranjang Sampah

Anda dapat menggunakan fitur berikut untuk memantau Keranjang Sampah.

Topik

- [Pantau Recycle Bin menggunakan Amazon EventBridge](#)
- [Monitor Recycle Bin menggunakan AWS CloudTrail](#)

Pantau Recycle Bin menggunakan Amazon EventBridge

Recycle Bin mengirimkan peristiwa ke Amazon EventBridge untuk tindakan yang dilakukan pada aturan retensi. Dengan EventBridge, Anda dapat menetapkan aturan yang memulai tindakan terprogram dalam menanggapi peristiwa ini. Misalnya, Anda dapat membuat EventBridge aturan yang mengirimkan pemberitahuan ke email Anda ketika aturan retensi dibuka dan memasuki periode penundaan buka kunci. Untuk informasi selengkapnya, lihat [Membuat EventBridge aturan Amazon yang bereaksi terhadap peristiwa](#).

Peristiwa di EventBridge direpresentasikan sebagai objek JSON. Bidang yang unik untuk peristiwa tersebut terdapat di bagian detail dari objek JSON. Bidang event berisi nama peristiwa. Bidang result berisi status selesai dari tindakan yang memulai peristiwa. Untuk informasi selengkapnya, lihat [pola EventBridge acara Amazon](#) di Panduan EventBridge Pengguna Amazon.

Untuk informasi selengkapnya tentang Amazon EventBridge, lihat [Apa itu Amazon EventBridge?](#) di Panduan EventBridge Pengguna Amazon.

Peristiwa

- [RuleLocked](#)
- [RuleChangeMencoba](#)
- [RuleUnlockDijadwalkan](#)
- [RuleUnlockingPemberitahuan](#)
- [RuleUnlocked](#)

RuleLocked

Berikut ini adalah contoh peristiwa yang dihasilkan Keranjang Sampah saat aturan retensi berhasil dikunci. Acara ini dapat dihasilkan oleh CreateRule dan LockRule permintaan. API yang menghasilkan peristiwa dicatat di bidang api-name.

```
{
  "version": "0",
  "id": "exampleb-b491-4cf7-a9f1-bf370example",
  "detail-type": "Recycle Bin Rule Locked",
  "source": "aws.rbin",
  "account": "123456789012",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
```

```

    "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
  ],
  "detail":
  {
    "detail-version": " 1.0.0",
    "rule-id": "a12345abcde",
    "rule-description": "locked account level rule",
    "unlock-delay-period": "30 days",
    "api-name": "CreateRule"
  }
}

```

RuleChangeMencoba

Berikut ini adalah contoh peristiwa yang dihasilkan Keranjang Sampah untuk upaya yang gagal untuk memodifikasi atau menghapus aturan terkunci. Acara ini dapat dihasilkan oleh DeleteRule dan UpdateRule permintaan. API yang menghasilkan peristiwa dicatat di bidang api-name.

```

{
  "version": "0",
  "id": "exampleb-b491-4cf7-a9f1-bf370example",
  "detail-type": "Recycle Bin Rule Change Attempted",
  "source": "aws.rbin",
  "account": "123456789012",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
  ],
  "detail":
  {
    "detail-version": " 1.0.0",
    "rule-id": "a12345abcde",
    "rule-description": "locked account level rule",
    "unlock-delay-period": "30 days",
    "api-name": "DeleteRule"
  }
}

```

RuleUnlockDijadwalkan

Berikut ini adalah contoh peristiwa yang dihasilkan Keranjang Sampah saat aturan retensi tidak terkunci dan memulai periode penundaan pembukaan kuncinya.

```
{
  "version": "0",
  "id": "exampleb-b491-4cf7-a9f1-bf370example",
  "detail-type": "Recycle Bin Rule Unlock Scheduled",
  "source": "aws.rbin",
  "account": "123456789012",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
  ],
  "detail": {
    "detail-version": " 1.0.0",
    "rule-id": "a12345abcde",
    "rule-description": "locked account level rule",
    "unlock-delay-period": "30 days",
    "scheduled-unlock-time": "2022-09-10T16:37:50Z",
  }
}
```

RuleUnlockingPemberitahuan

Berikut ini adalah contoh peristiwa yang dihasilkan Keranjang Sampah setiap harinya saat aturan retensi berada dalam periode penundaan pembukaan kunci, hingga sehari sebelum periode penundaan pembukaan kunci berakhir.

```
{
  "version": "0",
  "id": "exampleb-b491-4cf7-a9f1-bf370example",
  "detail-type": "Recycle Bin Rule Unlocking Notice",
  "source": "aws.rbin",
  "account": "123456789012",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"
  ],
  "detail": {
    "detail-version": " 1.0.0",
    "rule-id": "a12345abcde",
    "rule-description": "locked account level rule",
  }
}
```

```
"unlock-delay-period": "30 days",  
"scheduled-unlock-time": "2022-09-10T16:37:50Z"  
}  
}
```

RuleUnlocked

Berikut ini adalah contoh peristiwa yang dihasilkan Keranjang Sampah saat periode penundaan pembukaan kunci untuk aturan retensi berakhir dan aturan retensi dapat dimodifikasi atau dihapus.

```
{  
  "version": "0",  
  "id": "exampleb-b491-4cf7-a9f1-bf370example",  
  "detail-type": "Recycle Bin Rule Unlocked",  
  "source": "aws.rbin",  
  "account": "123456789012",  
  "time": "2022-08-10T16:37:50Z",  
  "region": "us-west-2",  
  "resources": [  
    "arn:aws:rbin:us-west-2:123456789012:rule/a12345abcde"  
  ],  
  "detail":  
  {  
    "detail-version": " 1.0.0",  
    "rule-id": "a12345abcde",  
    "rule-description": "locked account level rule",  
    "unlock-delay-period": "30 days",  
    "scheduled-unlock-time": "2022-09-10T16:37:50Z"  
  }  
}
```

Monitor Recycle Bin menggunakan AWS CloudTrail

Layanan Recycle Bin terintegrasi dengan AWS CloudTrail. CloudTrail adalah layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan. CloudTrail menangkap semua panggilan API yang dilakukan di Recycle Bin sebagai peristiwa. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon Simple Storage Service (Amazon S3). Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa manajemen terbaru di CloudTrail konsol dalam Riwayat acara. Anda dapat menggunakan informasi yang dikumpulkan oleh CloudTrail untuk menentukan permintaan

yang dibuat untuk Recycle Bin, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk informasi selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

Informasi Recycle Bin di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Ketika aktivitas acara yang didukung terjadi di Recycle Bin, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat [Melihat Acara dengan Riwayat CloudTrail Acara](#).

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk Recycle Bin, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua AWS Wilayah. Jejak mencatat peristiwa dari semua Wilayah di AWS partisi dan mengirimkan file log ke bucket S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat [Gambaran Umum tentang pembuatan jejak](#) di Panduan Pengguna AWS CloudTrail .

Tindakan API yang didukung

Untuk Recycle Bin, Anda dapat menggunakan CloudTrail untuk mencatat tindakan API berikut sebagai peristiwa manajemen.

- CreateRule
- UpdateRule
- GetRules
- ListRule
- DeleteRule
- TagResource
- UntagResource
- ListTagsForResource
- LockRule
- UnlockRule

Untuk informasi selengkapnya tentang peristiwa pengelolaan [logging](#), lihat [peristiwa manajemen logging untuk jejak](#) di Panduan CloudTrail Pengguna.

Informasi identitas

Setiap peristiwa atau entri log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut hal ini:

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna terfederasi.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat [CloudTrail pengguna IdentityElement](#).

Memahami entri file log Keranjang Sampah

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga file tersebut tidak muncul dalam urutan tertentu.

Berikut ini adalah contoh entri CloudTrail log.

CreateRule

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
```

```
    "userName": "Admin"
  },
  "webIdFederationData": {},
  "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2021-08-02T21:43:38Z"
  }
},
"eventTime": "2021-08-02T21:45:22Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "CreateRule",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 botocore/1.21.9",
"requestParameters": {
  "retentionPeriod": {
    "retentionPeriodValue": 7,
    "retentionPeriodUnit": "DAYS"
  },
  "description": "Match all snapshots",
  "resourceType": "EBS_SNAPSHOT"
},
"responseElements": {
  "identifier": "jkrnexample"
},
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

GetRule

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-08-02T21:43:38Z"
      }
    }
  },
  "eventTime": "2021-08-02T21:45:33Z",
  "eventSource": "rbin.amazonaws.com",
  "eventName": "GetRule",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
  "requestParameters": {
    "identifier": "jkrnexample"
  },
  "responseElements": null,
  "requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
  "eventID": "714fafex-2eam-42pl-913e-926d4example",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "123456789012",
  "tlsDetails": {
```

```

    "tlsVersion": "TLSv1.2",
    "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
    "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
  }
}

```

ListRules

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-08-02T21:43:38Z"
      }
    }
  },
  "eventTime": "2021-08-02T21:44:37Z",
  "eventSource": "rbin.amazonaws.com",
  "eventName": "ListRules",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
  "requestParameters": {
    "resourceTags": [
      {
        "resourceTagKey": "test",
        "resourceTagValue": "test"
      }
    ]
  }
}

```

```

    }
  ]
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

UpdateRule

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-08-02T21:43:38Z"
      }
    }
  }
},

```

```

"eventTime": "2021-08-02T21:46:03Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "UpdateRule",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 boto-core/1.21.9",
"requestParameters": {
  "identifier": "jkrnexample",
  "retentionPeriod": {
    "retentionPeriodValue": 365,
    "retentionPeriodUnit": "DAYS"
  },
  "description": "Match all snapshots",
  "resourceType": "EBS_SNAPSHOT"
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

DeleteRule

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {

```

```

    "type": "Role",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:role/Admin",
    "accountId": "123456789012",
    "userName": "Admin"
  },
  "webIdFederationData": {},
  "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2021-08-02T21:43:38Z"
  }
}
},
"eventTime": "2021-08-02T21:46:25Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "DeleteRule",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.9 Python/3.6.14
Linux/4.9.230-0.1.ac.224.84.332.metal1.x86_64 botocore/1.21.9",
"requestParameters": {
  "identifier": "jkrnexample"
},
"responseElements": null,
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
}

```

TagResource

```

{
  "eventVersion": "1.08",
  "userIdentity": {

```

```
"type": "AssumedRole",
"principalId": "123456789012",
"arn": "arn:aws:iam::123456789012:root",
"accountId": "123456789012",
"accessKeyId": "AKIAIOSFODNN7EXAMPLE",
"sessionContext": {
  "sessionIssuer": {
    "type": "Role",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:role/Admin",
    "accountId": "123456789012",
    "userName": "Admin"
  },
  "webIdFederationData": {},
  "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2021-10-22T21:38:34Z"
  }
}
},
"eventTime": "2021-10-22T21:43:15Z",
"eventSource": "rbin.amazonaws.com",
"eventName": "TagResource",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 boto-core/1.21.26",
"requestParameters": {
  "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234",
  "tags": [
    {
      "key": "purpose",
      "value": "production"
    }
  ]
}
},
"responseElements": null,
"requestID": "examplee-7962-49ec-8633-795efexample",
"eventID": "example4-6826-4c0a-bdec-0bab1example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
```

```
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

UntagResource

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-10-22T21:38:34Z"
      }
    },
    "webIdFederationData": {},
    "attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2021-10-22T21:38:34Z"
    }
  },
  "eventTime": "2021-10-22T21:44:16Z",
  "eventSource": "rbin.amazonaws.com",
  "eventName": "UntagResource",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 botocore/1.21.26",
  "requestParameters": {
    "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234",
    "tagKeys": [
      "purpose"
    ]
  }
}
```

```

    ]
  },
  "responseElements": null,
  "requestID": "example7-6c1e-4f09-9e46-bb957example",
  "eventID": "example6-75ff-4c94-a1cd-4d5f5example",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "123456789012",
  "tlsDetails": {
    "tlsVersion": "TLSv1.2",
    "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
    "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
  }
}

```

ListTagsForResource

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-10-22T21:38:34Z"
      }
    }
  },
  "eventTime": "2021-10-22T21:42:31Z",

```

```

"eventSource": "rbin.amazonaws.com",
"eventName": "ListTagsForResource",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.123.123.123",
"userAgent": "aws-cli/1.20.26 Python/3.6.14
Linux/4.9.273-0.1.ac.226.84.332.metal1.x86_64 boto-core/1.21.26",
"requestParameters": {
  "resourceArn": "arn:aws:rbin:us-west-2:123456789012:rule/ABCDEF01234"
},
"responseElements": null,
"requestID": "example8-10c7-43d4-b147-3d9d9example",
"eventID": "example2-24fc-4da7-a479-c9748example",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",
"recipientAccountId": "123456789012",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

LockRule

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},

```

```
    "attributes": {
      "creationDate": "2022-10-25T00:45:11Z",
      "mfaAuthenticated": "false"
    }
  },
  "eventTime": "2022-10-25T00:45:19Z",
  "eventSource": "rbin.amazonaws.com",
  "eventName": "LockRule",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "python-requests/2.25.1",
  "requestParameters": {
    "identifier": "jkrnexample",
    "lockConfiguration": {
      "unlockDelay": {
        "unlockDelayValue": 7,
        "unlockDelayUnit": "DAYS"
      }
    }
  },
  "responseElements": {
    "identifier": "jkrnexample",
    "description": "",
    "resourceType": "EBS_SNAPSHOT",
    "retentionPeriod": {
      "retentionPeriodValue": 7,
      "retentionPeriodUnit": "DAYS"
    }
  },
  "resourceTags": [],
  "status": "available",
  "lockConfiguration": {
    "unlockDelay": {
      "unlockDelayValue": 7,
      "unlockDelayUnit": "DAYS"
    }
  },
  "lockState": "locked"
},
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
```

```

"recipientAccountId": "123456789012",
"eventCategory": "Management",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}

```

UnlockRule

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2022-10-25T00:45:11Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2022-10-25T00:46:17Z",
  "eventSource": "rbin.amazonaws.com",
  "eventName": "UnlockRule",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "python-requests/2.25.1",
  "requestParameters": {
    "identifier": "jkrnexample"
  },
}

```

```
"responseElements": {
  "identifier": "jkrnexample",
  "description": "",
  "resourceType": "EC2_IMAGE",
  "retentionPeriod": {
    "retentionPeriodValue": 7,
    "retentionPeriodUnit": "DAYS"
  },
  "resourceTags": [],
  "status": "available",
  "lockConfiguration": {
    "unlockDelay": {
      "unlockDelayValue": 7,
      "unlockDelayUnit": "DAYS"
    }
  },
  "lockState": "pending_unlock",
  "lockEndTime": "Nov 1, 2022, 12:46:17 AM"
},
"requestID": "ex0577a5-amc4-pl4f-ef51-50fdexample",
"eventID": "714fafex-2eam-42pl-913e-926d4example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789012",
"eventCategory": "Management",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "clientProvidedHostHeader": "rbin.us-west-2.amazonaws.com"
}
}
```

Lokasi sumber daya

Sumber daya Amazon EC2 khusus untuk AWS Wilayah atau Zona Ketersediaan tempat mereka tinggal.

Sumber Daya	Tipe	Deskripsi
Pengidentifikasi sumber daya Amazon EC2	Wilayah	Setiap pengidentifikasi sumber daya, seperti ID AMI, ID instans, ID volume EBS, atau ID snapshot EBS, terikat dengan Wilayahnya dan hanya dapat digunakan di Wilayah tempat Anda membuat sumber daya tersebut.
Nama sumber daya yang diberikan pengguna	Wilayah	Setiap nama sumber daya, seperti nama grup keamanan atau nama pasangan kunci, terikat dengan Wilayahnya dan hanya dapat digunakan di Wilayah tempat Anda menciptakan sumber daya tersebut. Meskipun Anda dapat membuat sumber daya dengan nama yang sama di banyak Wilayah, sumber daya tersebut tidak saling berhubungan.
AMI	Wilayah	AMI terikat dengan Wilayah tempat file berada dalam Amazon S3. Anda dapat menyalin sebuah AMI dari satu Wilayah ke Wilayah lainnya. Untuk informasi selengkapnya, lihat Menyalin AMI .
Snapshot EBS	Wilayah	Snapshot EBS terikat dengan Wilayahnya dan hanya dapat digunakan untuk membuat volume di Wilayah yang sama. Anda dapat menyalin snapshot dari satu Wilayah ke Wilayah lainnya.
Volume EBS	Zona Ketersediaan	Volume Amazon EBS terikat dengan Zona Ketersediaanya dan hanya dapat dilampirkan pada instans di Zona Ketersediaan yang sama.
Alamat IP elastis	Wilayah	Alamat IP Elastis terikat dengan satu Wilayah dan hanya dapat dikaitkan dengan instans di Wilayah yang sama.
Instans	Zona Ketersediaan	Instans terikat dengan Zona Ketersediaan tempat Anda meluncurkannya. Akan tetapi, ID instansnya terikat dengan Wilayahnya.

Sumber Daya	Tipe	Deskripsi
Key pair	Global atau Regional	<p>Pasangan kunci yang Anda buat menggunakan Amazon EC2 terikat dengan Wilayah tempat Anda membuatnya. Anda dapat membuat pasangan kunci RSA Anda sendiri dan mengunggahnya ke Wilayah tempat Anda ingin menggunakannya; oleh karena itu, Anda dapat menjadikan pasangan kunci tersedia secara global dengan mengunggahnya ke setiap Wilayah.</p> <p>Untuk informasi selengkapnya, lihat Pasangan kunci Amazon EC2 dan instans Amazon EC2.</p>
Grup keamanan	Wilayah	<p>Grup keamanan terikat dengan satu Wilayah dan hanya dapat ditetapkan pada instans di Wilayah yang sama. Anda tidak dapat memungkinkan instans untuk berkomunikasi dengan instans di luar Wilayahnya menggunakan aturan grup keamanan. Lalu lintas dari satu instans di Wilayah lain akan terlihat sebagai bandwidth WAN.</p>

ID sumber daya

Saat sumber daya dibuat, kami menetapkan ID sumber daya yang unik untuk setiap sumber daya. ID sumber daya berbentuk seperti pengidentifikasi sumber daya (seperti snap untuk snapshot) yang diikuti dengan tanda hubung dan kombinasi unik dari huruf dan angka.

Setiap pengidentifikasi sumber daya, seperti ID AMI, ID instans, ID volume EBS, atau ID snapshot EBS, terikat dengan Wilayahnya dan hanya dapat digunakan di Wilayah tempat Anda membuat sumber daya tersebut.

Anda dapat menggunakan ID sumber daya untuk menemukan sumber daya di konsol Amazon EC2. Jika Anda menggunakan alat baris perintah atau API Amazon EC2 untuk menggunakan Amazon EC2, ID sumber daya diperlukan untuk perintah tertentu. Misalnya, jika Anda menggunakan AWS CLI perintah [stop-instance](#) untuk menghentikan instance, Anda harus menentukan ID instance dalam perintah.

Panjang ID sumber daya

Sebelum Januari 2016, ID yang ditetapkan ke sumber daya yang baru dibuat dari tipe sumber daya tertentu menggunakan 8 karakter setelah tanda hubung (misalnya, i-1a2b3c4d). Sejak Januari 2016 hingga Juni 2018, kami mengubah ID tipe sumber daya ini menjadi 17 karakter setelah tanda hubung (misalnya, i-1234567890abcdef0). Bergantung pada waktu pembuatan akun, Anda mungkin memiliki beberapa sumber daya yang sudah ada dengan ID pendek, tetapi, sumber daya baru akan memiliki ID yang lebih panjang.

Membuat daftar dan memfilter sumber daya 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](#)
- [Membuat daftar serta memfilter menggunakan CLI dan API](#)
- [Melihat sumber daya di seluruh Wilayah menggunakan Amazon EC2 Global View](#)

Membuat daftar dan memfilter sumber daya menggunakan konsol

Daftar Isi

- [Membuat daftar sumber daya menggunakan konsol](#)
- [Memfilter sumber daya menggunakan konsol](#)
 - [Filter yang didukung](#)

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/>.

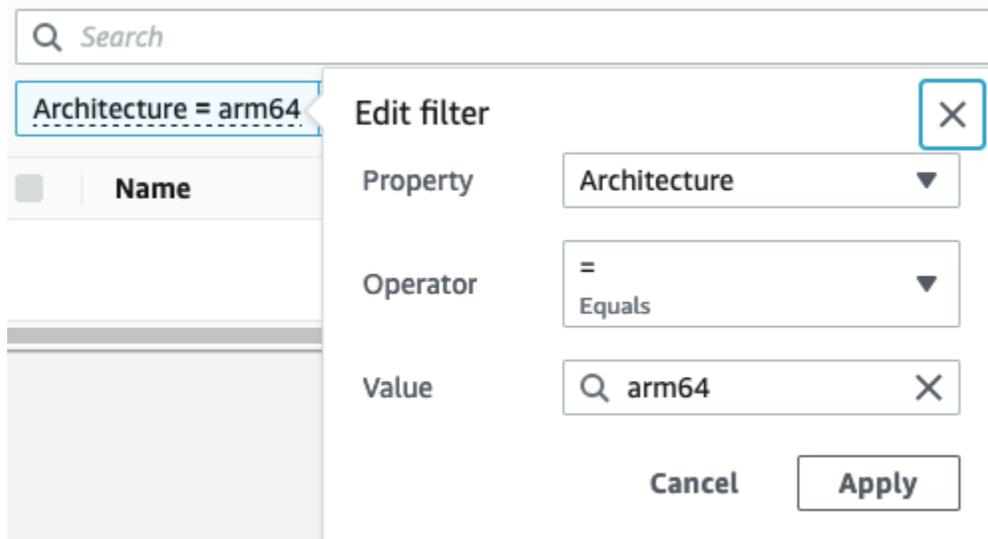
2. Di panel navigasi, pilih opsi yang sesuai dengan tipe sumber daya tersebut. Misalnya, untuk membuat daftar instans Anda, pilih Instans.

Halaman akan menampilkan semua sumber daya dari tipe sumber daya yang dipilih.

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 pada 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 dikosongkan, 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.

Pencarian peka huruf besar/kecil

Untuk menggunakan pencarian yang peka huruf besar/kecil, Anda harus memilih kotak centang Gunakan pencocokan peka huruf besar/kecil di jendela Preferensi. 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.

Membuat daftar serta memfilter menggunakan CLI dan API

Setiap tipe sumber daya memiliki perintah CLI dan tindakan API terkait yang dapat Anda gunakan untuk membuat daftar sumber daya dari tipe tersebut. 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 per filter dalam satu permintaan.

- String filter dapat mencapai 255 karakter panjangnya.
- 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?\`.

Filter yang didukung

Guna melihat filter yang didukung untuk setiap sumber daya Amazon EC2, lihat dokumentasi berikut:

- AWS CLI: Perintah `describe` di dalam [Referensi Perintah AWS CLI Amazon EC2](#).
- Alat untuk Windows PowerShell: Get Perintah dalam [Referensi AWS Tools for PowerShell Cmdlet-A Amazon EC2](#).
- API Kueri: Tindakan API `Describe` di dalam [Referensi API Amazon EC2](#).

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 perintah berikut untuk menyertakan instans yang berjalan dalam output saja.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running
```

Guna membuat daftar ID instans saja untuk instans berjalan Anda, tambahkan parameter `--query` sebagai berikut.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running --query "Reservations[*].Instances[*].InstanceId" --output text
```

Berikut ini adalah output contoh.

```
i-0ef1f57f78d4775a4  
i-0626d4edd54f1286d  
i-04a636d18e83cfacb
```

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 perintah berikut untuk membuat daftar semua instans dengan tipe `m5.large` atau `m5d.large`.

```
aws ec2 describe-instances --filters Name=instance-type,Values=m5.large,m5d.large
```

Anda dapat menggunakan perintah berikut untuk membuat daftar semua instans yang dihentikan dengan tipe `t2.micro`.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=stopped  
Name=instance-type,Values=t2.micro
```

Example Contoh: Gunakan wildcard dalam nilai filter

Jika Anda menentukan `database` sebagai nilai filter untuk filter `description` saat mendeskripsikan snapshot EBS menggunakan [describe-snapshots](#), perintah hanya akan menampilkan snapshot dengan deskripsi “basis data”.

```
aws ec2 describe-snapshots --filters Name=description,Values=database
```

Wildcard `*` cocok dengan nol karakter atau lebih. Jika Anda menentukan `*database*` sebagai nilai filter, perintah hanya akan menampilkan snapshot yang deskripsinya mencakup basis data kata.

```
aws ec2 describe-snapshots --filters Name=description,Values=*database*
```

Wildcard `?` cocok dengan 1 karakter saja. Jika Anda menentukan `database?` sebagai nilai filter, perintah hanya akan menampilkan snapshot dengan deskripsi “basis data” atau “basis data” yang diikuti satu karakter.

```
aws ec2 describe-snapshots --filters Name=description,Values=database?
```

Jika Anda menentukan `database????`, perintah hanya akan menampilkan snapshot dengan deskripsi “basis data” yang diikuti hingga empat karakter. Perintah ini tidak menyertakan deskripsi “basis data” yang diikuti lima karakter atau lebih.

```
aws ec2 describe-snapshots --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 dari semua snapshot yang dibuat oleh Anda Akun AWS (diwakili oleh 123456789012) sebelum tanggal yang ditentukan (diwakili oleh 2020-03-31).* Jika Anda tidak menentukan pemiliknya, hasilnya akan menyertakan semua snapshot publik.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]" --output text
```

Perintah berikut menampilkan ID dari semua snapshot yang dibuat dalam rentang tanggal tertentu.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime>='2019-01-01') && (StartTime<='2019-12-31')].[SnapshotId]" --output text
```

Memfilter berdasarkan tanda

Untuk contoh tentang cara memfilter daftar sumber daya menurut tandanya, lihat [Bekerja dengan tanda menggunakan baris perintah](#).

Melihat sumber daya di seluruh Wilayah menggunakan Amazon EC2 Global View

Amazon EC2 Global View memungkinkan Anda melihat dan mencari sumber daya Amazon EC2 dan Amazon VPC dalam satu Wilayah, atau di AWS beberapa Wilayah secara bersamaan dalam satu konsol. Untuk informasi selengkapnya, lihat [Amazon EC2 Global View](#).

Amazon EC2 Global View

Amazon EC2 Global View memungkinkan Anda melihat beberapa sumber daya Amazon EC2 dan Amazon VPC dalam satu Wilayah AWS, atau di banyak Wilayah dalam satu konsol. Amazon EC2 Global View juga menyediakan fungsionalitas pencarian global yang memungkinkan Anda mencari sumber daya tertentu atau tipe sumber daya tertentu di banyak Wilayah secara bersamaan.

Amazon EC2 Global View tidak memungkinkan Anda memodifikasi sumber daya dengan cara apa pun.

Sumber daya yang didukung

Menggunakan Amazon EC2 Global View, Anda dapat melihat ringkasan global sumber daya berikut di semua Wilayah tempat Anda Akun AWS diaktifkan.

- Grup Auto Scaling
- Pengaturan opsi DHCP
- Gateway internet khusus egress
- IP Elastis
- Layanan titik akhir
- Instans
- Gateway internet
- Daftar prefiks terkelola
- Gateway NAT
- ACL jaringan
- Antarmuka jaringan
- Tabel rute
- Grup keamanan
- Subnet
- Volume
- VPC
- Titik akhir VPC
- Koneksi peering VPC

Izin yang diperlukan

Pengguna harus memiliki izin berikut untuk menggunakan Amazon EC2 Global View.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
"Effect": "Allow",
"Action": [
"autoscaling:DescribeAutoScalingGroups",
"ec2:DescribeRegions",
"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"
],
"Resource": "*"
}]
}
```

Untuk menggunakan Amazon EC2 Global View

Buka konsol Amazon EC2 Global View di <https://console.aws.amazon.com/ec2globalview/home>.

 Important

Anda tidak dapat menggunakan jendela privat di Firefox untuk mengakses Amazon EC2 Global View.

Konsol tersebut terdiri dari hal-hal berikut:

- Penjelajah Wilayah—Tab ini mencakup bagian-bagian berikut:
 - Ringkasan—Menyediakan gambaran umum tingkat tinggi tentang sumber daya Anda di semua Wilayah.

Wilayah yang Diaktifkan menunjukkan jumlah Wilayah tempat Anda Akun AWS diaktifkan. 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.

- Jumlah wilayah sumber daya—Daftar semua Wilayah AWS (termasuk yang tidak diaktifkan oleh akun Anda) dan menyediakan total untuk setiap tipe 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—Tab ini memungkinkan Anda mencari sumber daya tertentu atau tipe sumber daya tertentu di satu Wilayah atau di banyak 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.

Tip

Jika hanya menggunakan Wilayah atau tipe sumber daya tertentu, Anda dapat menyesuaikan Amazon EC2 Global View agar hanya menampilkan Wilayah dan tipe sumber daya tersebut. Untuk menyesuaikan Wilayah dan tipe sumber daya yang ditampilkan, di panel navigasi, pilih Pengaturan, lalu pada tab Sumber Daya dan Wilayah, pilih Wilayah dan tipe sumber daya yang tidak ingin ditampilkan di Amazon EC2 Global View.

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 tanda](#)
- [Tandai sumber daya Anda](#)
- [Pembatasan tanda](#)
- [Manajemen tanda dan akses](#)
- [Menandai sumber daya Anda untuk penagihan](#)
- [Bekerja dengan tanda menggunakan konsol](#)
- [Bekerja dengan tanda menggunakan baris perintah](#)
- [Bekerja dengan tanda instans dalam metadata instans](#)
- [Tambahkan tag ke sumber daya menggunakan CloudFormation](#)

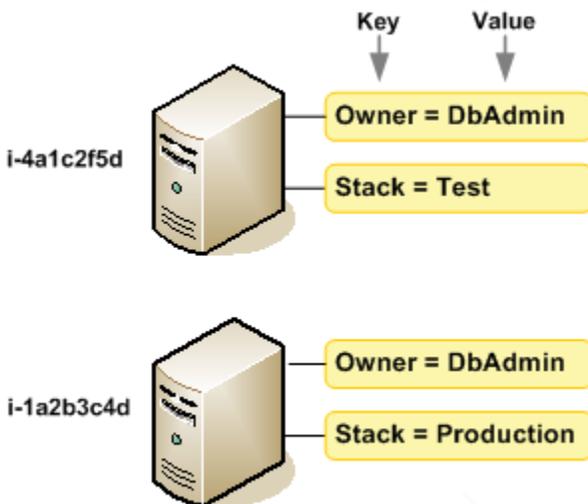
Dasar-dasar tanda

Tag adalah label yang Anda tetapkan ke AWS sumber daya. Setiap tanda terdiri dari kunci dan nilai opsional, yang keduanya Anda tentukan.

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 Anda merancang set kunci tanda yang memenuhi kebutuhan setiap tipe 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 [Praktik AWS Terbaik Tagging](#).

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

Anda dapat menandai sebagian besar sumber daya Amazon EC2 yang sudah ada dalam akun.

[Tabel](#) berikut mencantumkan sumber daya yang mendukung penandaan.

Jika menggunakan konsol Amazon EC2, Anda dapat menerapkan tag ke sumber daya menggunakan tab Tag di 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, the, atau AWS SDK AWS CLI, Anda dapat menggunakan tindakan `CreateTags` EC2 API 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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#).

Tabel berikut menjelaskan sumber daya Amazon EC2 yang dapat diberi tag, dan sumber daya yang dapat diberi tag saat pembuatan menggunakan Amazon EC2 API, the, atau SDK. AWS CLI AWS

Dukungan penandaan untuk sumber daya Amazon EC2

Sumber daya	Mendukung tag	Mendukung penandaan saat pembuatan
AFI	Ya	Ya
AMI	Ya	Ya
Tugas paket	Tidak	Tidak
Reservasi Kapasitas	Ya	Ya
Gateway pembawa	Ya	Ya
Titik akhir VPN klien	Ya	Ya
Rute VPN klien	Tidak	Tidak
Gateway pelanggan	Ya	Ya
Host Khusus	Ya	Ya
Reservasi Host Khusus	Ya	Ya
Opsi DHCP	Ya	Ya
Snapshot EBS	Ya	Ya
Volume EBS	Ya	Ya
Armada EC2	Ya	Ya
Gateway internet khusus egress	Ya	Ya
Alamat IP Elastis	Ya	Ya
Akselerator Elastic Graphics	Ya	Tidak
Instans	Ya	Ya

Sumber daya	Mendukung tag	Mendukung penandaan saat pembuatan
Jendela peristiwa instans	Ya	Ya
Volume penyimpanan instans	N/A	N/A
gateway internet	Ya	Ya
Kolam alamat IP (BYOIP)	Ya	Ya
Pasangan kunci	Ya	Ya
Templat peluncuran	Ya	Ya
Versi templat peluncuran	Tidak	Tidak
Gateway lokal	Ya	Tidak
Tabel rute gateway lokal	Ya	Tidak
Antarmuka virtual gateway lokal	Ya	Tidak
Grup antarmuka virtual gateway lokal	Ya	Tidak
Kaitan VPC tabel rute gateway lokal	Ya	Ya
Kaitan grup antarmuka virtual tabel rute gateway lokal	Ya	Tidak
Gateway NAT	Ya	Ya
ACL jaringan	Ya	Ya
Antarmuka jaringan	Ya	Ya
Grup penempatan	Ya	Ya

Sumber daya	Mendukung tag	Mendukung penandaan saat pembuatan
Daftar prefiks	Ya	Ya
Instans Terpesan	Ya	Tidak
Daftar Instans Terpesan	Tidak	Tidak
Tabel rute	Ya	Ya
Permintaan Armada Spot	Ya	Ya
Permintaan Instans Spot	Ya	Ya
Grup keamanan	Ya	Ya
Aturan grup keamanan	Ya	Tidak
Subnet	Ya	Ya
Filter Cermin Lalu Lintas	Ya	Ya
Sesi Cermin Lalu Lintas	Ya	Ya
Target Cermin Lalu Lintas	Ya	Ya
Gateway transit	Ya	Ya
Domain multicast gateway transit	Ya	Ya
Tabel rute gateway transit	Ya	Ya
Lampiran VPN gateway transit	Ya	Ya
Gateway privat virtual	Ya	Ya
VPC	Ya	Ya
Titik akhir VPC	Ya	Ya

Sumber daya	Mendukung tag	Mendukung penandaan saat pembuatan
Layanan titik akhir VPC	Ya	Ya
Konfigurasi layanan titik akhir VPC	Ya	Ya
Log aliran VPC	Ya	Ya
Koneksi peering VPC	Ya	Ya
Koneksi VPN	Ya	Ya

Anda dapat menandai instans, volume, grafik elastis, antarmuka jaringan, dan permintaan Instans Spot saat pembuatan menggunakan [wizard peluncuran instans](#) Amazon EC2 di konsol Amazon EC2. Anda dapat menandai volume EBS saat pembuatan menggunakan layar Volume, atau menandai snapshot EBS menggunakan layar Snapshot. Atau, gunakan API Amazon EC2 yang membuat sumber daya (misalnya [RunInstances](#),) untuk menerapkan tag saat membuat sumber daya Anda.

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 tanda:

- Jumlah maksimum tag per sumber daya – 50
- Untuk setiap sumber daya, setiap kunci tanda harus unik, dan setiap kunci tanda hanya dapat memuat satu nilai.
- Panjang kunci maksimum – 128 karakter Unicode dalam UTF-8
- Panjang nilai maksimum – 256 karakter Unicode dalam UTF-8
- Karakter yang diizinkan
 - Meskipun EC2 memungkinkan karakter apa pun dalam tagnya, AWS layanan lain lebih ketat. 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 [Bekerja dengan tanda instans dalam metadata instans](#).
- Kunci dan nilai tanda peka huruf besar/kecil.
- `aws` :Awalan 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.

Anda tidak dapat menandai semua sumber daya. Untuk informasi selengkapnya, lihat [Dukungan penandaan untuk sumber daya Amazon EC2](#).

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 [Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat](#).

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 [Mengendalikan akses ke sumber daya EC2 menggunakan tanda sumber daya](#).

Menandai sumber daya Anda untuk penagihan

Anda dapat menggunakan tag untuk mengatur AWS tagihan Anda untuk 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 menonaktifkan 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.

Bekerja dengan tanda menggunakan konsol

Anda dapat menggunakan konsol Amazon EC2 untuk menampilkan tanda sumber daya individual, serta menerapkan atau menghapus tanda dari satu sumber daya pada satu waktu.

Anda dapat menggunakan Editor Tag di AWS Resource Groups konsol untuk menampilkan tag dari semua sumber daya Amazon EC2 Anda di semua Wilayah. Anda dapat melihat tanda berdasarkan sumber daya dan tipe sumber daya, dan dapat melihat tipe sumber daya mana yang terkait dengan tanda tertentu. Anda dapat menerapkan atau menghapus tanda dari banyak sumber daya dan banyak tipe sumber daya sekaligus. Editor Tanda menyediakan cara terpusat dan terpadu untuk membuat dan mengelola tanda Anda. Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS Sumber Daya Penandaan](#).

Tugas

- [Tampilkan tanda](#)
- [Menambahkan dan menghapus tanda pada sumber daya individu](#)
- [Menambahkan dan menghapus tanda untuk banyak sumber daya](#)
- [Tambahkan tanda saat meluncurkan instans](#)
- [Memfilter daftar sumber daya berdasarkan tanda](#)

Tampilkan tanda

Anda dapat menampilkan tanda dari sumber daya individu di konsol Amazon EC2. Untuk menampilkan tanda dari semua sumber daya Anda, gunakan Editor Tanda di konsol AWS Resource Groups .

Menampilkan tanda dari sumber daya individu

Saat Anda memilih halaman khusus sumber daya dalam konsol Amazon EC2, halaman tersebut akan menampilkan daftar sumber daya. Misalnya, jika Anda memilih Instans dari panel navigasi, konsol akan menampilkan instans Amazon EC2 Anda. Saat Anda memilih sumber daya dari salah

satu daftar ini (misalnya, instans), jika sumber daya mendukung tanda, Anda dapat melihat dan mengelola tanda tersebut. Pada sebagian besar halaman sumber daya, Anda dapat melihat tanda dengan memilih tab Tanda.

Anda dapat menambahkan kolom ke daftar sumber daya untuk menampilkan semua nilai untuk tanda dengan kunci yang sama. Anda dapat menggunakan kolom ini untuk mengurutkan dan memfilter daftar sumber daya berdasarkan tandanya.

New console

Untuk menambahkan kolom ke daftar sumber daya guna menampilkan tanda Anda

1. Di konsol EC2, pilih ikon berbentuk roda gigi Preferensi di sudut kanan atas layar.
2. Di kotak dialog Preferensi, untuk Kolom tanda (di kiri bawah), pilih satu dari beberapa kunci tanda, lalu pilih Konfirmasi.

Old console

Ada dua cara untuk menambahkan kolom baru ke daftar sumber daya guna menampilkan tanda Anda:

- Pada tab Tanda, pilih Tampilkan Kolom. Kolom baru akan ditambahkan ke konsol.
- Pilih ikon berbentuk roda gigi Tampilkan/Sembunyikan Kolom, lalu dalam kotak dialog Tampilkan/Sembunyikan Kolom, pilih kunci tanda pada Kunci Tanda Anda.

Menampilkan tanda untuk banyak sumber daya

Anda dapat menampilkan tanda di banyak sumber daya menggunakan Editor Tanda di [konsol AWS Resource Groups](#). Untuk informasi selengkapnya, lihat [Panduan Pengguna AWS Sumber Daya Penandaan](#).

Menambahkan dan menghapus tanda pada sumber daya individu

Anda dapat mengelola tanda untuk sumber daya individu secara langsung dari halaman sumber daya tersebut.

Untuk menambahkan tanda ke sumber daya individu

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.

2. Dari bilah navigasi, pilih Wilayah tempat sumber daya yang akan ditandai berada. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
3. Di panel navigasi, pilih tipe sumber daya (misalnya, Instans).
4. Pilih sumber daya dari daftar sumber daya lalu pilih tab Tanda.
5. Pilih Kelola tanda, lalu pilih Tambahkan tanda baru. Masukkan kunci dan nilai untuk tanda tersebut. Pilih Tambahkan tanda baru lagi untuk setiap tanda tambahan yang akan ditambahkan. Setelah Anda selesai menambahkan tanda, pilih Simpan.

Untuk menghapus tanda dari sumber daya individu

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Dari bilah navigasi, pilih Wilayah tempat sumber daya yang akan dihapus tandanya berada. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
3. Di panel navigasi, pilih tipe sumber daya (misalnya, Instans).
4. Pilih sumber daya dari daftar sumber daya lalu pilih tab Tanda.
5. Pilih Kelola tanda. Untuk setiap tanda yang akan dihapus, pilih Hapus. Setelah Anda selesai menghapus tanda, pilih Simpan.

Menambahkan dan menghapus tanda untuk banyak sumber daya

Untuk menambahkan tanda ke banyak sumber daya

1. Buka Editor Tag di konsol AWS Resource Groups di <https://console.aws.amazon.com/resource-groups/tag-editor>.
2. Untuk Wilayah, pilih satu Wilayah atau lebih tempat sumber daya yang akan ditandai berada.
3. Untuk tipe Sumber Daya, pilih jenis sumber daya yang akan diberi tag (misalnya, AWS::EC2::Instance).
4. Pilih Cari sumber daya.
5. Pada Hasil pencarian sumber daya, pilih kotak centang yang berada di samping setiap sumber daya yang akan ditandai.
6. Pilih Kelola tanda sumber daya yang dipilih.
7. Pada Edit tanda dari semua sumber daya yang dipilih, pilih Tambahkan tanda, lalu masukkan kunci dan nilai tanda yang baru. Pilih Tambahkan tanda lagi untuk setiap tanda tambahan yang akan ditambahkan.

Note

Jika Anda menambahkan tanda baru yang memiliki kunci tanda yang sama dengan tanda yang sudah ada, tanda yang baru akan menimpa tanda yang sudah ada.

8. Pilih Tinjau dan terapkan perubahan tanda.
9. Pilih Terapkan perubahan ke semua yang dipilih.

Untuk menghapus tanda dari banyak sumber daya

1. Buka Editor Tag di konsol AWS Resource Groups di <https://console.aws.amazon.com/resource-groups/tag-editor>.
2. Untuk Wilayah, pilih Wilayah tempat sumber daya yang akan dihapus tandanya berada.
3. Untuk tipe Sumber Daya, pilih jenis sumber daya yang akan dihapus tag (misalnya, AWS::EC2::Instance).
4. Pilih Cari sumber daya.
5. Di bawah Hasil pencarian sumber daya, pilih kotak centang yang berada di samping setiap sumber daya yang akan dihapus tandanya.
6. Pilih Kelola tanda sumber daya yang dipilih.
7. Di bawah Edit tanda dari semua sumber daya yang dipilih, di samping tanda yang akan dihapus, pilih Hapus tanda.
8. Pilih Tinjau dan terapkan perubahan tanda.
9. Pilih Terapkan perubahan ke semua yang dipilih.

Tambahkan tanda saat meluncurkan instans

New console

Untuk menambahkan tanda menggunakan wizard peluncuran instans

1. Dari bilah navigasi, pilih Wilayah untuk instans tersebut. Pilihan ini penting karena beberapa sumber daya Amazon EC2 dapat dibagikan antar-Wilayah, sedangkan yang lainnya tidak. Pilih Wilayah yang sesuai dengan kebutuhan Anda. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).

2. Pilih Luncurkan instans.
3. Pada Nama dan tanda, Anda dapat memasukkan nama deskriptif untuk instans dan menentukan tanda.

Nama instans adalah tanda, di mana kuncinya adalah Nama, dan nilainya adalah nama yang Anda tentukan. Anda dapat menandai instans, volume, grafik elastis, dan antarmuka jaringan. Untuk Instans Spot, Anda hanya dapat menandai permintaan Instans Spot.

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 Tambahkan tanda lagi untuk setiap tanda tambahan yang akan ditambahkan.
4. Di bawah Citra Aplikasi dan OS (Amazon Machine Image), pilih sistem operasi (OS) untuk instans dan AMI Anda. Untuk informasi selengkapnya, lihat [Aplikasi dan Gambar OS \(Gambar Mesin Amazon\)](#).
 5. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang sudah ada atau buat yang baru.
 6. Simpan semua bidang lain pada nilai defaultnya atau pilih nilai tertentu untuk konfigurasi instans yang Anda inginkan. Untuk informasi tentang bidang tersebut, lihat [Luncurkan sebuah instans menggunakan parameter yang ditentukan](#).
 7. Di panel Ringkasan, tinjau pengaturan Anda, lalu pilih Luncurkan instans.

Old console

Untuk menambahkan tanda menggunakan wizard peluncuran instans

1. Dari bilah navigasi, pilih Wilayah untuk instans tersebut. Pilihan ini penting karena beberapa sumber daya Amazon EC2 dapat dibagikan antar-Wilayah, sedangkan yang lainnya tidak. Pilih Wilayah yang sesuai dengan kebutuhan Anda. Untuk informasi selengkapnya, lihat [Lokasi sumber daya](#).
2. Pilih Luncurkan Instans.

3. Halaman Pilih Amazon Machine Image (AMI) akan menampilkan daftar konfigurasi dasar yang disebut Amazon Machine Image (AMI). Pilih AMI yang ingin digunakan lalu pilih Pilih. Untuk informasi selengkapnya, lihat [Temukan AMI](#).
4. Pada halaman Konfigurasi Detail Instans, konfigurasi pengaturan instans sesuai kebutuhan, lalu pilih Berikutnya: Tambahkan Penyimpanan.
5. Pada halaman Tambahkan Penyimpanan, Anda dapat menentukan volume penyimpanan tambahan untuk instans. Pilih Berikutnya: Tambahkan Tanda setelah selesai.
6. Pada halaman Tambahkan Tanda, tentukan tanda untuk instans, volume, atau keduanya. Pilih Tambahkan tanda lain untuk menambahkan lebih dari satu tanda ke instans Anda. Pilih Berikutnya: Konfigurasi Grup Keamanan setelah Anda selesai.
7. Pada halaman Konfigurasi Grup Keamanan, Anda dapat memilih dari grup keamanan milik Anda yang sudah ada, atau membiarkan wizard membuat sebuah grup keamanan yang baru untuk Anda. Pilih Tinjau dan Luncurkan setelah Anda selesai.
8. Meninjau pengaturan Anda. Setelah yakin dengan pilihan Anda, pilih Luncurkan. Pilih pasangan kunci yang sudah ada atau buat yang baru, pilih kotak centang pengakuan, lalu pilih Luncurkan Instans.

Memfilter daftar sumber daya berdasarkan tanda

Anda dapat memfilter daftar sumber daya berdasarkan satu atau beberapa kunci tanda dan nilai tanda.

Untuk memfilter daftar sumber daya berdasarkan tanda di konsol Amazon EC2

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 [Membuat daftar dan memfilter sumber daya Anda](#).

Untuk memfilter banyak sumber daya di banyak Wilayah berdasarkan tanda menggunakan Editor Tanda

Anda dapat menggunakan Editor Tag di konsol AWS Resource Groups untuk memfilter beberapa sumber daya di beberapa Wilayah berdasarkan tag. Untuk informasi selengkapnya, lihat [Menemukan sumber daya yang akan ditandai](#) di Panduan Pengguna Penandaan Sumber Daya AWS .

Bekerja dengan tanda menggunakan baris perintah

Anda dapat menambahkan tanda ke banyak sumber daya EC2 saat membuatnya menggunakan parameter spesifikasi tanda untuk perintah create. Anda dapat melihat tanda untuk sebuah sumber daya menggunakan perintah describe untuk sumber daya tersebut. Anda juga dapat menambahkan, memperbarui, atau menghapus tanda untuk sumber daya yang sudah ada menggunakan perintah berikut.

Tugas	AWS CLI	AWS Tools for Windows PowerShell
Menambahkan atau menimpa satu tanda atau lebih	create-tags	New-EC2Tag
Menghapus satu tanda atau lebih	delete-tags	Remove-EC2Tag
Mendeskripsikan satu tanda atau lebih	describe-tags	Get-EC2Tag

Tugas

- [Menambahkan tanda pada pembuatan sumber daya](#)
- [Tambahkan tanda ke sumber daya yang ada](#)
- [Mendeskripsikan sumber daya yang ditandai](#)

Menambahkan tanda pada pembuatan sumber daya

Contoh berikut menunjukkan cara menerapkan tanda saat Anda membuat sumber daya.

Note

Cara memasukkan parameter yang berformat JSON pada baris perintah berbeda-beda tergantung sistem operasi Anda.

- Linux, macOS, atau Unix dan Windows PowerShell - Gunakan tanda kutip tunggal (') untuk melampirkan struktur data JSON.
- Windows – Hilangkan tanda kutip tunggal saat menggunakan perintah dengan baris perintah Windows.

Untuk informasi selengkapnya, lihat [Menentukan nilai parameter untuk AWS CLI](#).

Example Contoh: Luncurkan instans dan terapkan tanda ke instans serta volume

Perintah [run-instances](#) berikut ini akan meluncurkan sebuah instans dan menerapkan sebuah tanda dengan kunci **webserver** dan nilai **production** pada instans tersebut. Perintah ini juga menerapkan sebuah tanda dengan kunci **cost-center** dan nilai **cc123** pada volume EBS yang dibuat (dalam hal ini, volume root).

```
aws ec2 run-instances \  
  --image-id ami-abc12345 \  
  --count 1 \  
  --instance-type t2.micro \  
  --key-name MyKeyPair \  
  --subnet-id subnet-6e7f829e \  
  --tag-specifications  
'ResourceType=instance,Tags=[{Key=webserver,Value=production}]'  
'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Anda dapat menerapkan kunci dan nilai tanda yang sama pada instans serta volume selama peluncuran. Perintah berikut meluncurkan instans dan menerapkan sebuah tanda dengan kunci **cost-center** dan nilai **cc123** pada instans serta volume EBS yang dibuat.

```
aws ec2 run-instances \  
  --image-id ami-abc12345 \  
  --count 1 \  
  --instance-type t2.micro \  
  --key-name MyKeyPair \  
  --subnet-id subnet-6e7f829e \  
  --tag-specifications 'ResourceType=instance,Tags=[{Key=cost-center,Value=cc123}]'  
'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Example Contoh: Membuat volume dan menerapkan tanda

Perintah [create-volume](#) berikut ini akan membuat sebuah volume dan menerapkan dua tanda: **purpose=production** dan **cost-center=cc123**.

```
aws ec2 create-volume \  
  --availability-zone us-east-1a \  
  --volume-type gp2 \  
  --size 80 \  
  --tag-specifications 'ResourceType=volume,Tags=[{Key=purpose,Value=production},  
{Key=cost-center,Value=cc123}]'
```

Tambahkan tanda ke sumber daya yang ada

Contoh berikut menunjukkan cara menambahkan tanda 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-78a54011 \  
  --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-1a2b3c4d 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 melarikan diri dari karakter khusus, lampirkan nilai yang memiliki karakter khusus dengan tanda kutip ganda ("), mendahului setiap karakter kutipan ganda dengan garis miring terbalik (\), dan kemudian lampirkan seluruh kunci dan struktur nilai dengan tanda kutip tunggal (') sebagai berikut: '

```
aws ec2 create-tags `\  
  --resources i-1234567890abcdef0 `\  
  --tags 'Key="\[Group]",Value=test'
```

Mendeskripsikan sumber daya yang ditandai

Contoh berikut menunjukkan cara menggunakan filter dengan [describe-instances](#) untuk melihat instans dengan tanda tertentu. Semua perintah describe EC2 menggunakan sintaksis ini untuk memfilter berdasarkan tanda pada satu tipe sumber daya. Atau, Anda dapat menggunakan perintah [describe-tags](#) untuk memfilter berdasarkan tanda pada tipe sumber daya EC2.

Example Contoh: Mendeskripsikan instans dengan kunci tanda tertentu

Perintah berikut menjelaskan instans dengan sebuah tanda **Stack**, dengan tidak memedulikan nilai tanda tersebut.

```
aws ec2 describe-instances \  
  --filters Name=tag-key,Values=Stack
```

Example Contoh: Mendeskripsikan instans dengan tanda tertentu

Perintah berikut mendeskripsikan instans dengan tanda **Stack=production**.

```
aws ec2 describe-instances \  
  --filters Name=tag:Stack,Values=production
```

Example Contoh: Mendeskripsikan instans dengan nilai tanda tertentu

Perintah berikut mendeskripsikan instans dengan tanda **production**, terlepas dari kunci tandanya.

```
aws ec2 describe-instances \  
  --filters Name=tag-value,Values=production
```

Example Contoh: Mendeskripsikan semua sumber daya EC2 dengan tanda tertentu

Perintah berikut menjelaskan semua sumber daya EC2 dengan tanda **Stack=Test**.

```
aws ec2 describe-tags \  
  --filters Name=key,Values=Stack Name=value,Values=Test
```

Bekerja dengan tanda instans dalam 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.

Topik

- [Mengizinkan akses ke tanda dalam metadata instans](#)
- [Menonaktifkan akses ke tanda dalam metadata instans](#)
- [Lihat apakah akses ke tanda dalam metadata instans diizinkan](#)
- [Mengambil tanda dari metadata instans](#)

Mengizinkan akses ke tanda dalam metadata instans

Secara default, tidak ada akses ke tanda instans dalam metadata instans. Untuk setiap instans, Anda harus secara eksplisit mengizinkan akses menggunakan salah satu metode berikut.

Untuk mengizinkan akses ke tanda dalam metadata instans menggunakan konsol

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 mengizinkan akses ke tanda dalam metadata instans, pilih kotak centang Izinkan.
5. Pilih Simpan.

Untuk mengizinkan akses ke tanda dalam metadata instans saat peluncuran menggunakan AWS CLI

Gunakan perintah [run-instances](#) dan atur InstanceMetadataTags menjadi `enabled`.

```
aws ec2 run-instances \  
  --image-id ami-0abcdef1234567890 \  
  --instance-type c3.large \  
  ...  
  --metadata-options "InstanceMetadataTags=enabled"
```

Untuk mengizinkan akses ke tanda dalam metadata instans pada instans yang sedang berjalan atau berhenti menggunakan AWS CLI

Gunakan perintah [modify-instance-metadata-options](#) dan atur `--instance-metadata-tags` menjadi `enabled`.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-123456789example \  
  --instance-metadata-tags enabled
```

Menonaktifkan akses ke tanda dalam metadata instans

Untuk menonaktifkan akses ke tanda instans dalam metadata instans, gunakan salah satu metode berikut. Anda tidak perlu menonaktifkan akses ke tanda instans pada metadata instans saat peluncuran karena akses akan dinonaktifkan secara default.

Untuk menonaktifkan akses ke tanda dalam metadata instans menggunakan konsol

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 tanda dalam metadata instans, kosongkan kotak centang Izinkan.
5. Pilih Simpan.

Untuk mematikan akses ke tag dalam metadata contoh menggunakan AWS CLI

Gunakan perintah [modify-instance-metadata-options](#) dan atur `--instance-metadata-tags` menjadi `disabled`.

```
aws ec2 modify-instance-metadata-options \  
  --instance-id i-123456789example \  
  --instance-metadata-tags disabled
```

Lihat apakah akses ke tanda dalam metadata instans diizinkan

Untuk setiap instance, Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk melihat apakah akses ke tag instans dari metadata instans diizinkan.

Untuk melihat apakah akses ke tanda dalam metadata instans diizinkan menggunakan konsol

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi, pilih Instans, lalu pilih instans Anda.
3. Pada tab Detail, periksa bidang Izinkan tanda di metadata instans. Jika nilainya adalah Diaktifkan, tanda dalam metadata instans diizinkan. Jika nilainya Dinonaktifkan, tanda dalam metadata instans tidak diizinkan.

Untuk melihat apakah akses ke tag dalam metadata contoh diizinkan menggunakan AWS CLI

Gunakan perintah [describe-instances](#) dan tentukan ID instans.

```
aws ec2 describe-instances \  
  --instance-ids i-1234567890abcdef0
```

Contoh output berikut dipotong untuk ruang. Parameter "InstanceMetadataTags" menunjukkan apakah tanda dalam metadata instans diizinkan. Jika nilainya adalah `enabled`, tanda dalam metadata instans diizinkan. Jika nilainya adalah `disabled`, tanda dalam metadata instans tidak diizinkan.

```
{  
  "Reservations": [  
    {  
      "Groups": [],  
      "Instances": [  
        {  
          "AmiLaunchIndex": 0,  
          "ImageId": "ami-0abcdef1234567890",  
          "InstanceId": "i-1234567890abcdef0",  
          ...  
        }  
      ],  
      "MetadataOptions": {  
        "State": "applied",  
        "HttpTokens": "optional",  
        "HttpPutResponseHopLimit": 1,  
        "HttpEndpoint": "enabled",  
        "HttpProtocolIpv6": "disabled",  
        "InstanceMetadataTags": "enabled"  
      },  
      ...  
    }  
  ]  
}
```

Mengambil tanda dari metadata instans

Jika tanda instans diizinkan dalam metadata instans, kategori `tags/instance` dapat diakses dari metadata instans. Untuk contoh tentang cara mengambil tanda dari metadata instans, lihat [Dapatkan tanda instans untuk sebuah instans](#).

Tambahkan tag ke sumber daya menggunakan CloudFormation

Dengan tipe sumber daya Amazon EC2, Anda menentukan tanda menggunakan Tags atau properti `TagSpecifications`.

Contoh berikut menambahkan tag **Stack=Production** untuk [AWS::EC2::Instance](#) menggunakan Tags propertinya.

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 `TagSpecifications` propertinya.

Example Contoh: `TagSpecifications` di YAMAL

```
TagSpecifications:
  - ResourceType: "instance"
    Tags:
      - Key: "Stack"
        Value: "Production"
```

Example Contoh: TagSpecifications di JSON

```
"TagSpecifications": [  
  {  
    "ResourceType": "instance",  
    "Tags": [  
      {  
        "Key": "Stack",  
        "Value": "Production"  
      }  
    ]  
  }  
]
```

Kuota layanan Amazon EC2

Amazon EC2 menyediakan berbagai sumber daya yang dapat Anda gunakan. Sumber daya ini mencakup citra, instans, volume, dan snapshot. Saat Anda membuat Akun AWS, kami menetapkan kuota default (juga disebut sebagai batas) pada sumber daya ini berdasarkan per wilayah. Misalnya, ada jumlah instans maksimum yang dapat Anda luncurkan dalam sebuah Wilayah. Jadi, jika Anda ingin meluncurkan instans dalam Wilayah AS Barat (Oregon), misalnya, permintaan tidak boleh menyebabkan penggunaan Anda melebihi jumlah instans maksimum dalam Wilayah tersebut.

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.

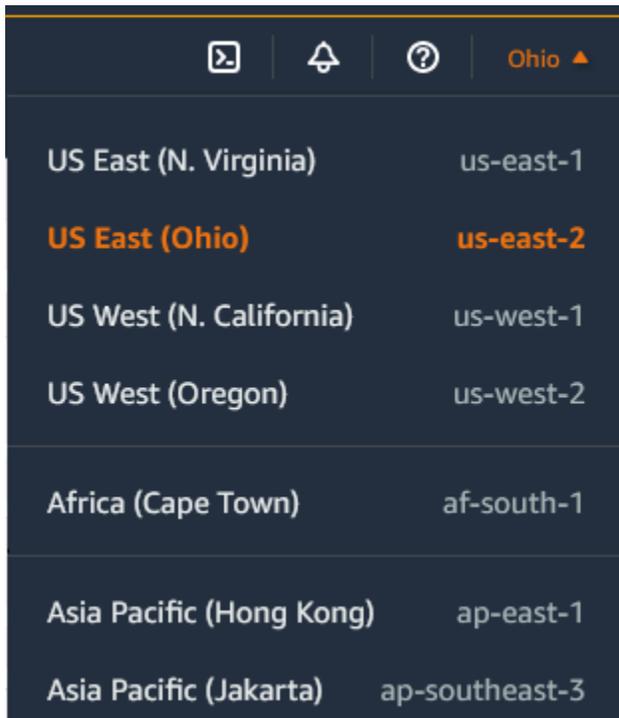
Untuk informasi selengkapnya, lihat titik akhir [dan kuota Amazon EC2 serta titik akhir](#) dan [kuota Amazon EBS](#) di. Referensi Umum Amazon Web

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 Kuota Layanan di <https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/>.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah.



Region	Code
US East (N. Virginia)	us-east-1
US East (Ohio)	us-east-2
US West (N. California)	us-west-1
US West (Oregon)	us-west-2
Africa (Cape Town)	af-south-1
Asia Pacific (Hong Kong)	ap-east-1
Asia Pacific (Jakarta)	ap-southeast-3

3. Gunakan bidang filter untuk memfilter daftar berdasarkan nama sumber daya. Misalnya, masukkan **On-Demand** guna menemukan kuota untuk Instans Sesuai Permintaan.
4. 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 Kuota Layanan 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** guna menemukan kuota untuk Instans Sesuai Permintaan.
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 AWS Support. 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

Di semua instans, Amazon EC2 membatasi lalu lintas keluar ke alamat IP publik melalui port 25 secara default. 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?](#)

Note

Pembatasan ini tidak berlaku untuk lalu lintas keluar yang dikirim melalui port 25 ke:

- 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](#).

Memecahkan masalah instans EC2

Prosedur dan tips berikut dapat membantu Anda memecahkan masalah dengan instans Amazon EC2 Anda.

Daftar Isi

- [Masalah umum dengan instans Windows](#)
- [Pesan umum dengan instance Windows](#)
- [Pemecahan masalah peluncuran instans](#)
- [Memecahkan masalah saat menghubungkan ke instans Linux](#)
- [Pemecahan masalah koneksi ke instans Windows Anda](#)
- [Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa](#)
- [Memecahkan masalah instans yang tidak dapat dijangkau](#)
- [Pemecahan masalah penghentian instans Anda](#)
- [Memecahkan masalah penghentian instans \(mematikan\)](#)
- [Memecahkan masalah instance Linux dengan pemeriksaan status yang gagal](#)
- [Memecahkan masalah booting instance Linux dari volume yang salah](#)
- [Memecahkan masalah Sysprep dengan instance Windows](#)
- [Penggunaan EC2Rescue untuk Linux](#)
- [Gunakan EC2Rescue untuk Windows Server](#)
- [Konsol Serial EC2 untuk instans Amazon EC2](#)
- [Kirimkan interupsi diagnostik \(untuk pengguna tingkat lanjut\)](#)

Masalah umum dengan instans Windows

Berikut ini adalah kiat-kiat pemecahan masalah untuk membantu Anda memecahkan masalah umum dengan instans Windows Server EC2.

Masalah

- [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](#)

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
```

Agan 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.

Versi Driver	DSRM Didukung?	Langkah Berikutnya
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 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, lihat [the section called “Adaptor Jaringan Elastis \(ENA\)”](#). Untuk informasi tentang memutakhirkan driver AWS PV, lihat [Memutakhirkan driver PV di instans Windows](#).

Konfigurasi 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, 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 mengatur kunci `RealTimeIsUniversal` 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 diatur ke `1: HKEY_LOCAL_MACHINE\SYSTEM\Set\Control\Information\CurrentControl TimeZone RealTime IsUniversal`

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 “Konfigurasi 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](#) () atau ()AWS CLI. [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 instance Windows Server 2012 R2 yang tidak tersedia di jaringan, lihat [Windows Server 2012 R2 kehilangan konektivitas jaringan dan penyimpanan setelah instance reboot](#).

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. Jalankan `select disk` DiskPart 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. Pada panel navigasi sebelah kiri, klik dua kali 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 ditunjukkan pada langkah sebelumnya tidak cocok dengan tanda tangan disk yang Anda tulis sebelumnya, gunakan DiskPart perintah berikut untuk mengubah tanda tangan disk agar cocok:

```
DISKPART> uniqueid disk id=E9EB3AA5
```

Pesan umum dengan instance Windows

Bagian ini berisi kiat-kiat untuk membantu Anda memecahkan masalah berdasarkan pesan umum.

Pesan

- [“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 AMI yang didukung Amazon EBS-Backed](#).

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 [Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa](#).

“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 [Cara menggunakan metadata instans](#).

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.) 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 ditunjukkan 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 RedHat 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. Buka <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 untuk 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. Temukan WindowsActivate plugin Ec2 di config.xml file. 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 fungsi Add-Routes untuk melihat daftar rute baru:

```
PS C:\> Add-Routes
```

3. Panggil ActivationSettings fungsi Set-:

```
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.

- Verifikasi bahwa 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
```

- Verifikasi bahwa kunci AWS KMS klien disetel. Jalankan perintah berikut dan periksa output-nya.

```
C:\Windows\System32\slmgr.vbs /dlv
```

Jika output berisi Error: 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: [Kunci Pengaturan AWS KMS Klien](#), 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 sudah 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”.

Pemecahan masalah peluncuran instans

Masalah berikut mencegah Anda meluncurkan instans.

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 yang tinggi segera setelah Windows dimulai \(hanya instance Windows\)](#)

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_id --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 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 terhenti, saat ini, AWS tidak memiliki kapasitas Sesuai Permintaan 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 [Ubah tipe instans](#).

- Jika Anda meluncurkan instans ke grup penempatan klaster, Anda bisa mendapatkan kesalahan kapasitas yang tidak memadai. Untuk informasi selengkapnya, lihat [Bekerja dengan grup penempatan](#).

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 [Batasan volume instans](#).
- 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.
- AMI yang didukung penyimpanan instans dan yang Anda gunakan untuk meluncurkan instans tidak memiliki bagian yang diperlukan (file image.part.xx).

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 Command Line Interface

1. Gunakan perintah [describe-instances](#) dan tentukan ID instans.

```
aws ec2 describe-instances --instance-id instance_id
```

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 peristiwa dengan riwayat peristiwa](#) 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 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 yang tinggi segera setelah Windows dimulai (hanya instance Windows)

Note

Tip pemecahan masalah ini hanya untuk instance 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 wuauclt net start wuauclt
```

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 instans Windows Anda saat diluncurkan](#).

Memecahkan masalah saat menghubungkan ke instans 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: Server menolak kunci kami atau Tidak tersedia metode autentikasi yang didukung](#)
- [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 caranya terhubung ke instans Linux 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 pada instans Linux Anda](#).

- Dapatkan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda:

AMI digunakan untuk meluncurkan instans	Nama pengguna default
AL2023	<code>ec2-user</code>
Amazon Linux 2	
Amazon Linux	
CentOS	<code>centos</code> atau <code>ec2-user</code>
Debian	<code>admin</code>
Fedora	<code>fedora</code> atau <code>ec2-user</code>
RHEL	<code>ec2-user</code> atau <code>root</code>
SEGAR	<code>ec2-user</code> atau <code>root</code>
Ubuntu	<code>ubuntu</code>
Oracle	<code>ec2-user</code>
Bitnami	<code>bitnami</code>
Linux Rocky	<code>rocky</code>
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 instans, perlu beberapa menit sampai instans tersebut siap, sehingga Anda dapat terhubung dengannya. 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 instans Anda telah lulus dua pemeriksaan status.

Verifikasi bahwa Anda telah memenuhi semua prasyarat untuk terhubung

Pastikan Anda memiliki semua informasi yang dibutuhkan untuk terhubung. Untuk informasi selengkapnya, lihat [Hubungkan ke instans Linux Anda](#).

Untuk prasyarat khusus tipe koneksi, seperti SSH, EC2 Instance Connect, OpenSSH, PuTTY, dan lainnya, lihat opsi berikut.

Linux atau macOS X

Jika sistem operasi komputer lokal Anda adalah Linux atau macOS X, periksa prasyarat khusus untuk opsi koneksi berikut:

- [Klien SSH](#)
- [EC2 Instance Connect](#)
- [AWS Systems Manager Manajer Sesi](#)

Windows

Jika sistem operasi komputer lokal Anda adalah Windows, periksa prasyarat khusus untuk opsi koneksi berikut:

- [OpenSSH](#)
- [PuTTY](#)

- [AWS Systems Manager Manajer Sesi](#)
- [Subsistem Windows untuk Linux](#)

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.
 - Untuk instans Linux: Verifikasi bahwa ada aturan yang mengizinkan lalu lintas dari komputer lokal Anda ke port 22 (SSH).
 - Untuk instans Windows: Verifikasi bahwa ada aturan yang mengizinkan lalu lintas dari komputer lokal Anda ke port 3389 (RDP).

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-xxxxxxx) 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 masuk dari alamat IP lokal Anda di port 22 (untuk instans Linux) atau port 3389 (untuk instans Windows). 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 kepada administrator jaringan apakah firewall internal mengizinkan lalu lintas masuk dan keluar dari komputer Anda di port 22 (untuk instans Linux) atau port 3389 (untuk instans Windows).

Jika komputer Anda memiliki firewall, verifikasi bahwa firewall mengizinkan lalu lintas masuk dan keluar dari komputer Anda di port 22 (untuk instans Linux) atau port 3389 (untuk instans Windows).

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 IP elastis](#).

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 [Pantau 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 [Ubah tipe instans](#).

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 pada port yang tepat (22 untuk Linux dan 3389 untuk Windows).
- 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 caranya terhubung ke instans Linux 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 instans Linux Anda dari Windows dengan 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 Nama host di dalam jendela Konfigurasi PuTTY.

AMI yang digunakan untuk meluncurkan instans	Nama pengguna default
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos atau ec2-user
Debian	admin
Fedora	fedora atau ec2-user
RHEL	ec2-user atau root
SEGAR	ec2-user atau root
Ubuntu	ubuntu
Oracle	ec2-user
Bitnami	bitnami
Linux Rocky	rocky

AMI yang digunakan untuk meluncurkan instans

Nama pengguna default

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 terhubung ke instans 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 privat dengan benar (file `.pem`) untuk instans Anda).

Nama pengguna yang sesuai adalah sebagai berikut:

AMI yang digunakan untuk meluncurkan instans	Nama pengguna default
AL2023 Amazon Linux 2 Amazon Linux	ec2-user
CentOS	centos atau ec2-user
Debian	admin
Fedora	fedora atau ec2-user
RHEL	ec2-user atau root
SEGAR	ec2-user atau root

AMI yang digunakan untuk meluncurkan instans	Nama pengguna default
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 caranya terhubung ke instans Linux 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 [Hentikan dan mulai 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 terhubung dari Windows, lakukan langkah-langkah berikut pada komputer lokal Anda.

1. Navigasikan ke file `.pem` Anda.
2. Klik kanan pada file `.pem` dan pilih Properti.
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 dapat terhubung ke instans Linux dari Windows melalui 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 dapat terhubung ke instans Linux dari Windows melalui 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: Server menolak kunci kami atau Tidak tersedia metode autentikasi yang didukung

Jika Anda menggunakan PuTTY untuk terhubung ke instans dan mendapatkan kesalahan berikut, Kesalahan: Server menolak kunci kami atau Kesalahan: Tidak tersedia metode autentikasi yang didukung, verifikasi bahwa Anda terhubung dengan nama pengguna yang sesuai untuk AMI Anda. Ketik nama pengguna pada Nama pengguna di jendela Konfigurasi PuTTY.

Nama pengguna yang sesuai adalah sebagai berikut:

AMI yang digunakan untuk meluncurkan instans	Nama pengguna default
AL2023	ec2-user
Amazon Linux 2	
Amazon Linux	
CentOS	centos atau ec2-user
Debian	admin
Fedora	fedora atau 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

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 instans Linux Anda dari Windows dengan 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 instance Anda, 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 CLI EC2 Instance Connect 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 default 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 dari Linux menggunakan 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 CLI EC2 Instance Connect 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 default-nya 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 dari Linux menggunakan 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 caranya terhubung ke instans Linux saya?

Jika Anda kehilangan kunci privat untuk instans yang didukung EBS, 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 [Siklus hidup instans](#).

Prosedur ini hanya didukung untuk instans dengan volume root EBS. Jika perangkat root adalah volume penyimpanan instans, Anda tidak dapat menggunakan prosedur ini untuk mendapatkan kembali akses ke instans Anda; Anda harus memiliki kunci privat untuk tersambung ke instans. Untuk menentukan tipe perangkat root instans Anda, buka konsol Amazon EC2, pilih Instans, pilih instans, pilih tab Penyimpanan, dan di bagian Detail perangkat root, periksa nilai Tipe 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 terhubung ke instans yang didukung EBS 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 instans 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, baik instans sudah dihentikan maupun perangkat root-nya adalah volume penyimpanan instans.

Warning

Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan instans, pastikan untuk mencadangkannya ke penyimpanan persisten.

Langkah 4: Luncurkan instans sementara

New console

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 AMI yang didukung Amazon EBS-Backed](#).
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.

Old console

Pilih Luncurkan instans, lalu gunakan wizard peluncuran untuk meluncurkan instans sementara dengan opsi berikut:

- Pada halaman Pilih AMI, 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 AMI yang didukung Amazon EBS-Backed](#).
- Pada halaman Pilih Tipe Instans, tinggalkan tipe instans default yang dipilih oleh wizard untuk Anda.
- Pada halaman Konfigurasi Detail Instans, tentukan Zona Ketersediaan yang sama dengan instans asli. Jika Anda meluncurkan instans dalam VPC, pilih sebuah subnet di Zona Ketersediaan ini.
- Pada halaman Tambahkan Tanda, tambahkan tanda Name=Temporary ke instans untuk mengindikasikan bahwa ini adalah instans sementara.
- Pada halaman Tinjau, pilih Luncurkan. Pilih pasangan kunci yang Anda buat di Langkah 1, lalu pilih Luncurkan Instans.

Langkah 5: Copot volume root dari instans asli dan lampirkan ke instans sementara

1. Di panel navigasi, pilih Volume dan pilih volume perangkat root untuk instans asli (Anda sudah 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 terlebih dahulu menghentikan instans sementara sebelum Anda dapat melampirkan volume.

Langkah 6: Tambahkan kunci publik baru ke **authorized_keys** pada volume asli yang dipasang ke instans 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 instans 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 output contoh ini, **222** adalah ID pengguna dan **500** adalah 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. Copot volume dari instans sementara (Anda melepasnya di langkah sebelumnya): Dari konsol Amazon EC2, pilih Volume di panel navigasi, pilih volume perangkat root untuk instans asli (Anda sudah mencatat ID volume pilih di langkah sebelumnya), 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 instans dari instans asli, tentukan nama perangkat yang Anda catat sebelumnya di [Langkah 2](#) untuk lampiran perangkat root asli (`/dev/sda1` atau `/dev/xvda`), lalu pilih Lampirkan volume.

⚠ Important

Jika Anda tidak menentukan nama perangkat yang sama dengan lampiran asli, Anda tidak dapat memulai instans asli. Amazon EC2 memperkirakan volume perangkat 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 instans sementara, dan pilih Status instans, Akhiri instans.

Pemecahan masalah koneksi ke instans Windows Anda

Informasi berikut dan kesalahan umum dapat membantu Anda memecahkan masalah saat menghubungkan ke instance 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 IP elastis](#).
- 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 mengizinkan akses RDP.
- 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 Anda](#) dan [the section called "Pemeriksaan status gagal di Linux"](#).
- 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. Anda dapat menggunakan [AWSSupport-TroubleshootRDP](#) untuk [disable the Windows Firewall profiles using SSM Agent](#). Untuk menonaktifkan Windows Firewall pada instance Windows yang tidak dikonfigurasi untuk AWS Systems Manager [AWSSupport-ExecuteEC2Rescue](#), menggunakan, atau menggunakan langkah-langkah manual berikut:

Langkah-langkah manual

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. 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\SharedAccess\Parameters\FirewallPolicy. Untuk setiap kunci dengan nama formulir xxxxProfile, pilih kunci dan ubah EnableFirewall dari 1 menjadi 0. Pilih lagi kunci tersebut, dan dari menu File, pilih Bongkar Hive.
6. (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 ditunjukkan 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
```

7. 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.

8. Lepaskan volume dari instans sementara. Anda dapat mengakhiri instans sementara jika tidak menggunakannya lagi.

9. Pulihkan volume root dari instans yang terpengaruh dengan melampirkannya sebagai /dev/sda1.

10. Mulai instans.

- Verifikasi bahwa Autentikasi Tingkat Jaringan dinonaktifkan pada instans yang bukan bagian dari domain Direktori Aktif (gunakan [AWSSupport-TroubleshootRDP](#) untuk [disable NLA](#)).
- Verifikasi bahwa Remote Desktop Service (TermService) Jenis Startup Otomatis dan layanan dimulai (gunakan [AWSSupport-TroubleshootRDP](#) untuk [enable 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 [Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa](#).
- 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 terhubung 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. Untuk informasi selengkapnya, lihat [Lingkungan Sesi Jarak Jauh](#).
- 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 digunakan secara berlebihan, server mungkin 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 [Ubah tipe instans](#).

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

Dokumen otomatisasi AWSSupport -TroubleShootrDP 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.

Dokumen otomatisasi AWSSupport -TroubleShootrDP dapat digunakan dengan instans EC2, instans lokal, dan mesin virtual (VM) yang diaktifkan untuk digunakan dengan (instance terkelola). AWS Systems Manager 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 AWS Systems Manager, lihat [Node terkelola](#) di AWS Systems Manager Panduan Pengguna.

Untuk memecahkan masalah menggunakan dokumen AWSSupport -TroubleShootrDP

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 pemilih 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 dari langkah-langkah individu, di Langkah-langkah yang Dieksekusi, pilih item dari ID Langkah.

AWSSupportContoh -TroubleshootRDP

Contoh berikut menunjukkan cara menyelesaikan tugas pemecahan masalah umum menggunakan -TroubleShootrDP. AWSSupport Anda dapat menggunakan AWS CLI [start-automation-execution](#) perintah contoh atau tautan yang disediakan ke file AWS Management Console.

Example Contoh: Periksa status RDP saat ini

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --parameters "InstanceId=instance_id, 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=instance_id, 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=instance_id, 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=instance_id, 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=instance_id, 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=instance_id, 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-EC2

Dokumen otomatisasi AWSSupport -ExecuteEC2Rescue menggunakan EC2Rescue untuk Windows Server untuk secara otomatis memecahkan masalah dan memulihkan konektivitas instans EC2 dan masalah RDP. Untuk informasi selengkapnya, lihat [Jalankan alat EC2Rescue di instans yang tak terjangkau](#).

Dokumen otomatisasi AWSSupport -ExecuteEC2Rescue 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 dokumen -ExecuteEC2Rescue AWSSupport

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 input, untuk UnreachableInstanceid, masukkan ID instans Amazon EC2 dari instance yang tidak dapat dijangkau.
7. (Opsional) Untuk LogDestination, masukkan nama bucket Amazon Simple Storage Service (Amazon S3) Simple Storage 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 Systems Manager Session Manager, 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. Klik kanan volume yang baru dilampirkan, yang berasal dari instans tak terjangkau, 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 Registry Editor, pilih hive 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, navigasikan ke HKEY_LOCAL_MACHINE*your key name*\ControlSet001\Control\Terminal Server, lalu di panel detail, klik dua kali fDenyTSConnections.
 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 Registry Editor, pilih HKEY_LOCAL_MACHINE*your-key-name*, lalu pilih File, Bongkar 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 [Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa](#). 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 .

Atur ulang kata sandi administrator Windows yang hilang atau kedaluwarsa

Note

Bagian ini hanya berlaku untuk instance Windows.

Jika Anda tidak dapat mengakses instans Amazon EC2 Windows lagi 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

- [Atur ulang kata sandi administrator Windows menggunakan EC2Launch v2](#)
- [Atur ulang kata sandi administrator Windows menggunakan EC2Config](#)
- [Atur ulang kata sandi administrator Windows menggunakan EC2Launch](#)

Atur ulang kata sandi administrator Windows 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 [Atur ulang kata sandi administrator Windows menggunakan EC2Launch](#).

Jika Anda menggunakan AMI Windows Server atau versi yang lebih baru dari Windows Server 2016 dan tidak menyertakan agen EC2Launch v2, lihat [Atur ulang kata sandi administrator Windows 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 file .run-once](#)
- [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. Pilih Tindakan, Citra dan templat, Buat citra. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar. Di panel navigasi, pilih AMI. Setelah status gambar berubah menjadi tersedia, lanjutkan ke langkah berikutnya.
 - d. Pilih gambar lalu pilih Tindakan, lalu Luncurkan.
 - e. Lengkapi wizard, memilih tipe instans, VPC, subnet, grup keamanan, dan peran IAM yang sama dengan instans yang digantikan, lalu pilih Luncurkan.
 - f. Saat diminta, pilih pasangan kunci yang Anda buat untuk instans baru, pilih kotak centang persetujuan, lalu pilih Luncurkan Instans.
 - 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 Storage. 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 instans 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 file .run-once

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](#).

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 [Siklus hidup instans](#).

4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

Atur ulang kata sandi administrator Windows 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 [Atur ulang kata sandi administrator Windows 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. (Konsol baru) Pilih Tindakan, Pantau dan pecahkan masalah, Dapatkan log sistem.

(Konsol lama) Pilih Tindakan, Pengaturan Sistem, Dapatkan Log Sistem.
4. Temukan entri Agen EC2, misalnya Agen EC2: 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. Pilih Tindakan, Citra dan templat, Buat citra. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar. Di panel navigasi, pilih AMI. Setelah status gambar berubah menjadi tersedia, lanjutkan ke langkah berikutnya.
 - d. Pilih gambar lalu pilih Tindakan, lalu Luncurkan.
 - e. Lengkapi wizard, memilih tipe instans, VPC, subnet, grup keamanan, dan peran IAM yang sama dengan instans yang digantikan, lalu pilih Luncurkan.
 - f. Saat diminta, pilih pasangan kunci yang Anda buat untuk instans baru, pilih kotak centang persetujuan, lalu pilih Luncurkan Instans.
 - 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 Storage. 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 instans 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.

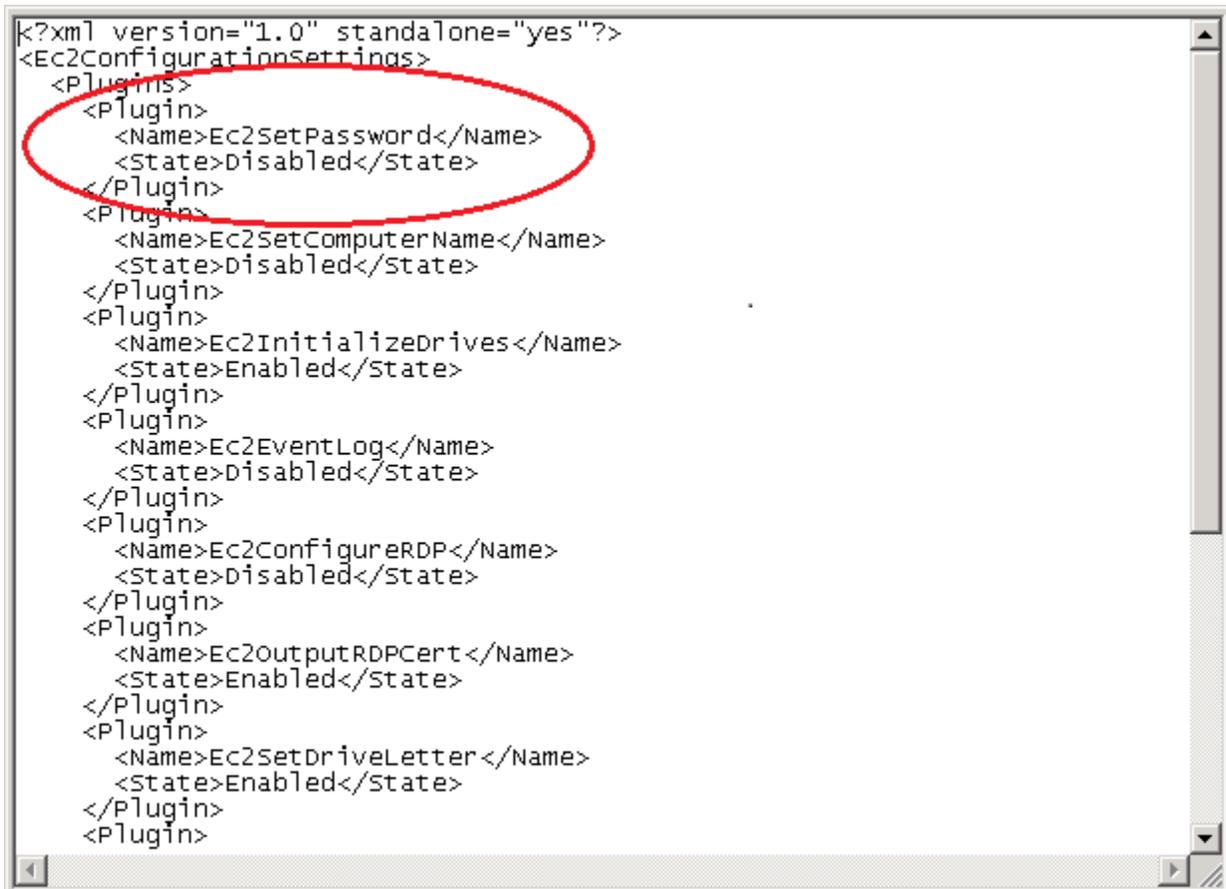
Langkah 4: Ubah file konfigurasi

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.



```
<?xml version="1.0" standalone="yes"?>
<Ec2ConfigurationSettings>
  <Plugins>
    <Plugin>
      <Name>Ec2SetPassword</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetComputerName</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2InitializeDrives</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2EventLog</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2ConfigureRDP</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2OutputRDPcert</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetDriveLetter</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
  
```

2. Setelah mengubah file konfigurasi, copot volume sekunder dari instans sementara dengan cara seperti berikut:
 - a. Menggunakan utilitas 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](#).

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 [Siklus hidup instans](#).

4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

Atur ulang kata sandi administrator Windows 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 [Atur ulang kata sandi administrator Windows menggunakan EC2Config](#).

Warning

Ketika Anda menghentikan suatu instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menjaga data dari volume penyimpanan instans, pastikan untuk mencadangkannya 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. Pilih Tindakan, Citra dan templat, Buat citra. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar. Di panel navigasi, pilih AMI. Setelah status gambar berubah menjadi tersedia, lanjutkan ke langkah berikutnya.
 - d. Pilih gambar lalu pilih Tindakan, lalu Luncurkan.
 - e. Lengkapi wizard, memilih tipe instans, VPC, subnet, grup keamanan, dan peran IAM yang sama dengan instans yang digantikan, lalu pilih Luncurkan.
 - f. Saat diminta, pilih pasangan kunci yang Anda buat untuk instans baru, pilih kotak centang persetujuan, lalu pilih Luncurkan Instans.
 - 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 Storage. 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 instans 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 for Windows Server](#), ekstrak kontennya, lalu 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](#).
4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

Memecahkan masalah instans yang tidak dapat dijangkau

Anda dapat menggunakan metode berikut untuk memecahkan masalah instans Amazon EC2 yang tidak dapat dijangkau.

Daftar Isi

- [Boot ulang instans](#)
- [Output konsol instans](#)
- [Mengambil tangkapan layar instans yang tidak dapat dijangkau](#)
- [Tangkapan layar umum untuk instance Windows](#)
- [Pemulihan instans saat komputer host gagal](#)

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 [Menyalakan ulang instans 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.

- Instance 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 mencakup tiga kesalahan log peristiwa sistem terakhir.

Secara opsional, Anda dapat mengambil output konsol serial terbaru kapan saja selama siklus hidup instans. Opsi ini hanya didukung pada [instance yang dibangun di Sistem AWS Nitro](#). Opsi ini tidak didukung melalui konsol Amazon EC2.

Note

Hanya output terbaru sebesar 64 KB dari yang di-posting itu disimpan, yang tersedia setidaknya 1 jam setelah posting terakhir.

Hanya pemilik instans yang dapat mengakses output konsol.

Gunakan salah satu metode berikut untuk mendapatkan output konsol.

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, lalu pilih Tindakan, Pantau dan pecahkan masalah, Dapatkan log sistem.

Command line

Untuk mendapatkan output konsol

Anda dapat menggunakan salah satu perintah berikut ini. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [get-console-output](#) (AWS CLI)
- [Get-EC2ConsoleOutput](#) (AWS Tools for Windows PowerShell)

Mengambil tangkapan layar instans yang tidak dapat dijangkau

Jika Anda tidak dapat terhubung ke instans Anda, Anda dapat mengambil tangkapan layar dari 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 yang didukung oleh prosesor Graviton berbasis ARM](#)
- Instance Windows aktif AWS Outposts

Wilayah yang didukung

Fitur ini tersedia di Wilayah berikut:

- Wilayah AS Timur (Virginia Utara)
- Wilayah US East (Ohio)
- Wilayah US West (N California)
- Wilayah US West (Oregon)
- Wilayah Afrika (Cape Town)
- Wilayah Asia Pasifik (Hong Kong)
- Wilayah Asia Pasifik (Hyderabad)
- Wilayah Asia Pasifik (Jakarta)

- Wilayah Asia Pasifik (Melbourne)
- Wilayah Asia Pasifik (Mumbai)
- Wilayah Asia Pasifik (Osaka)
- Wilayah Asia Pacific (Seoul)
- Wilayah Asia Pasifik (Singapura)
- Wilayah Asia Pasifik (Sydney)
- Wilayah Asia Pasifik (Tokyo)
- Wilayah Kanada (Pusat)
- Wilayah Kanada Barat (Calgary)
- Wilayah Tiongkok (Beijing)
- Wilayah Tiongkok (Ningxia)
- Wilayah Eropa (Frankfurt)
- Wilayah Eropa (Irlandia)
- Wilayah Eropa (London)
- Wilayah Eropa (Milan)
- Wilayah Eropa (Paris)
- Wilayah Eropa (Spanyol)
- Wilayah Eropa (Stockholm)
- Wilayah Eropa (Zürich)
- Wilayah Israel (Tel Aviv)
- Wilayah Amerika Selatan (Sao Paulo)
- Wilayah Timur Tengah (Bahrain)
- Wilayah Timur Tengah (UEA)

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 klik kanan gambar untuk mengunduh dan menyimpannya.

Command line

Untuk mengambil tangkapan layar suatu instans

Anda dapat menggunakan salah satu perintah berikut ini. Konten yang dikembalikan adalah diberi kode base64. Untuk informasi selengkapnya tentang antarmuka baris perintah ini, lihat [Akses Amazon EC2](#).

- [get-console-screenshot](#) (AWS CLI)
- [GetConsoleTangkapan layar](#) (API Kueri Amazon EC2)

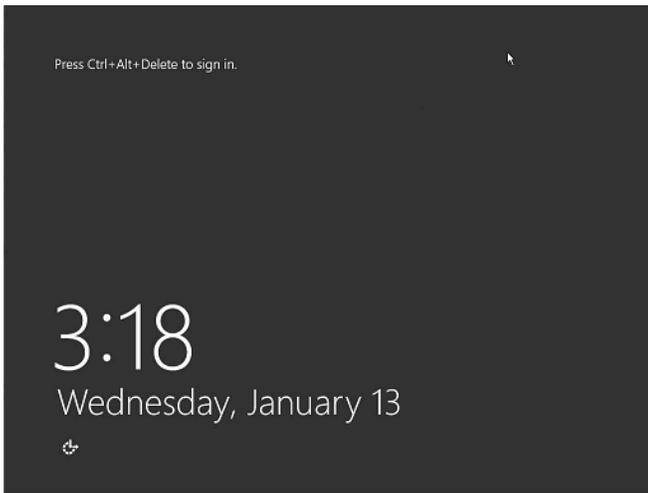
Tangkapan layar umum untuk instance Windows

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 instans.

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 instans tidak terkait dengan IP Elastic, IP publik berubah setelah instans berhenti/dimulai. 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

Konfigurasi	Verifikasi
	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 Bagaimana cara memecahkan masalah konektivitas tunnel VPN ke Amazon VPC?

Konfigurasi jaringan Windows

Konfigurasi	Verifikasi
Windows Firewall	Verifikasi bahwa Windows Firewall tidak memblokir koneksi ke instans Anda. Nonaktifkan 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 .
Konfigurasi TCP/IP 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 Management Console (MMC) (<code>services.msc</code>). 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 Terminal seperti yang dijelaskan dalam Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh .

Konfigurasi	Verifikasi
	Setelah Anda selesai, lampirkan kembali volume root ke instans asli.

Penggunaan CPU yang tinggi

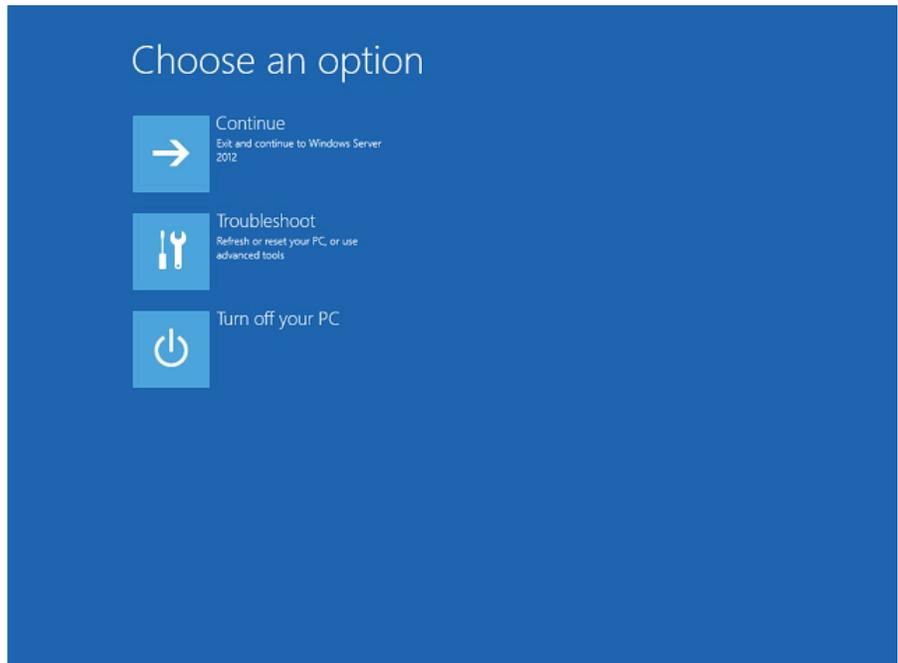
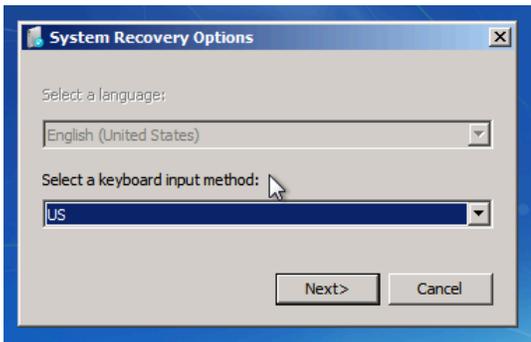
Periksa metrik CPUUtilization (Maximum) pada instans Anda dengan menggunakan Amazon CloudWatch. Jika CPUUtilization (Maksimum) menampilkan angka yang besar, tunggu hingga angka CPU turun dan coba sambungkan lagi. Penggunaan CPU yang tinggi dapat disebabkan oleh:

- Pembaruan Windows
- Pemindaian Perangkat Lunak Keamanan
- Skrip Startup Kustom
- Penjadwal Tugas

Untuk informasi selengkapnya, lihat [Mendapatkan Statistik untuk Sumber Daya Tertentu](#) di Panduan CloudWatch Pengguna Amazon. Untuk kiat-kiat pemecahan masalah tambahan, lihat [Penggunaan CPU yang tinggi segera setelah Windows dimulai \(hanya instance Windows\)](#).

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 ke `ignoreallfailures`.

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.

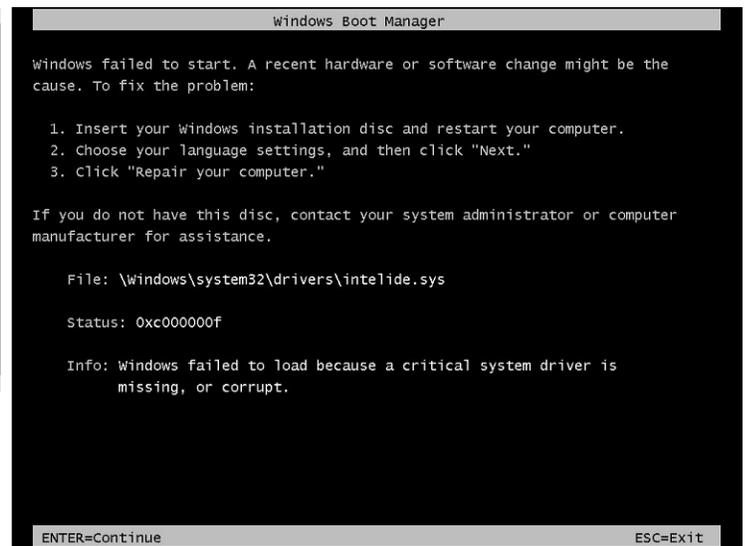
3. Masuk ke instans dan jalankan perintah berikut dari prompt perintah untuk mengubah konfigurasi `bootstatuspolicy` ke `ignoreallfailures`.

```
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

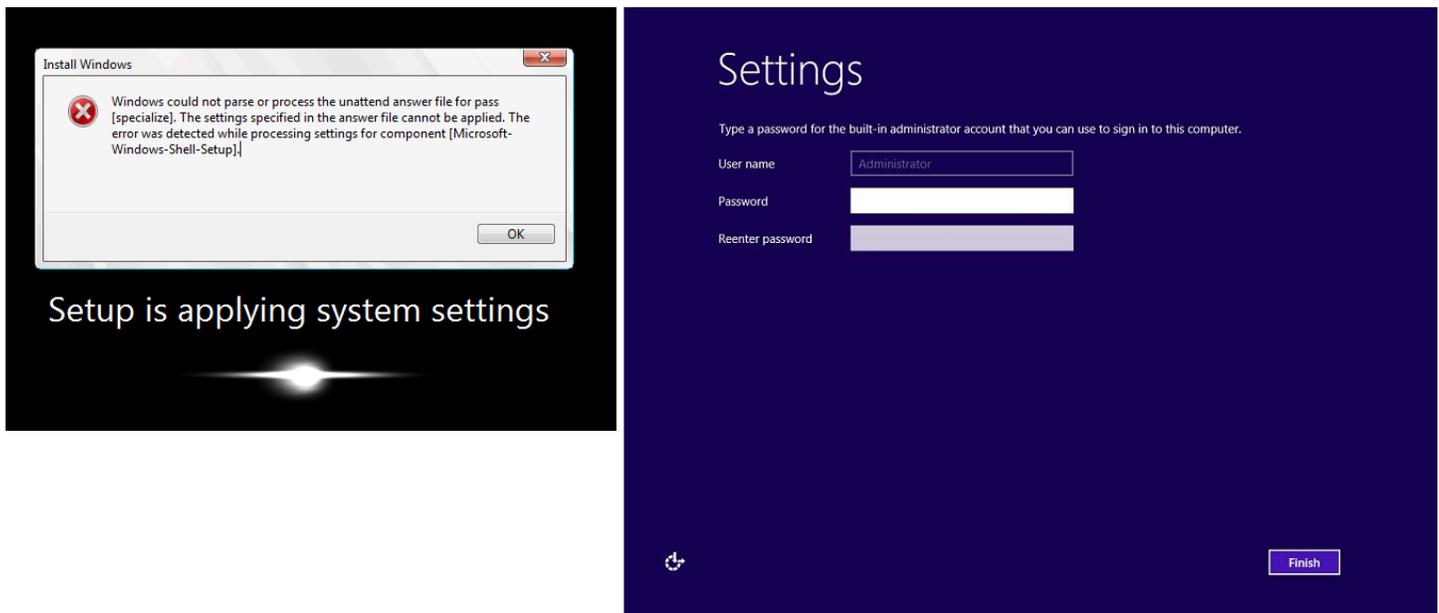
Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.



Sistem operasi mengalami kerusakan fatal pada file sistem dan/atau registri. 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

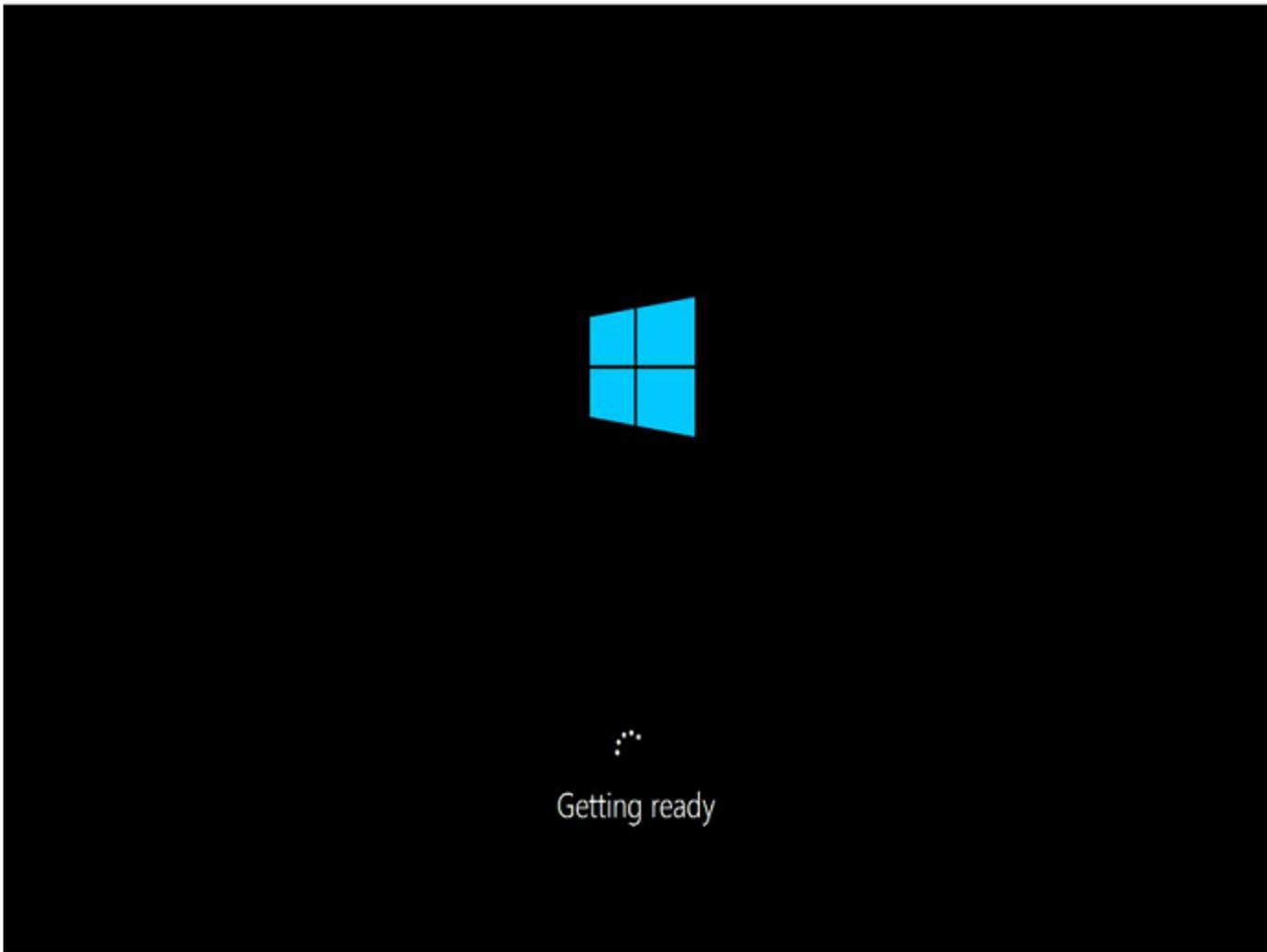
Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.



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 dengan 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 CPU Utilization (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.



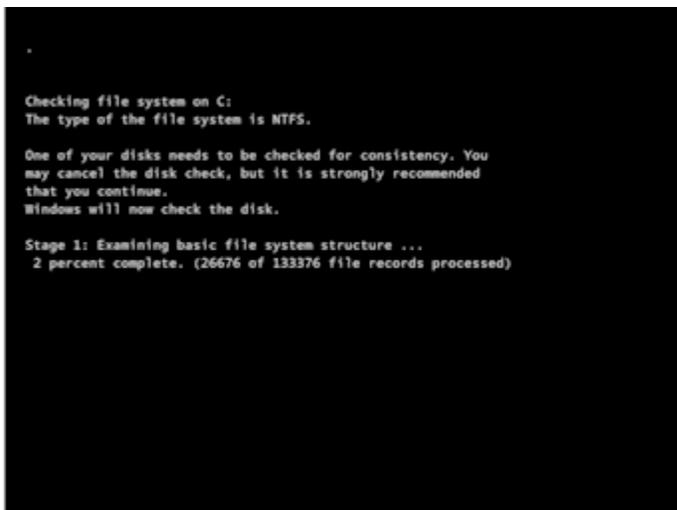
Proses Pembaruan Windows sedang memperbarui registri. Tunggu hingga pembaruan selesai. Jangan lakukan boot ulang atau penghentian instans karena ini dapat menyebabkan kerusakan data selama pembaruan.

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 instans yang didukung Amazon EBS 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 [Hentikan dan mulai instans Amazon EC2](#).

Untuk memulihkan instans yang didukung penyimpanan instans 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.

Pemecahan masalah penghentian instans Anda

Jika Anda telah menghentikan instans yang didukung Amazon EBS dan instans terlihat macet dalam status `stopping`, mungkin ada masalah dengan komputer host yang mendasarinya.

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`.

Hentikan paksa instans

Hentikan paksa instans menggunakan konsol atau AWS CLI.

Note

Anda dapat memaksa instans untuk berhenti menggunakan konsol hanya saat instans dalam status `stopping`. Anda dapat memaksa instans untuk berhenti menggunakan AWS CLI saat instans dalam status apa pun, kecuali `shutting-down` dan `terminated`.

Console

Untuk menghentikan paksa instans dengan menggunakan konsol

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, Hentikan paksa instan, Berhenti.

Perhatikan bahwa Penghentian paksa instans hanya tersedia di konsol jika instans Anda dalam status `stopping`. Jika instance Anda dalam keadaan lain (kecuali `shutting-down` dan `terminated`), Anda dapat menggunakan AWS CLI untuk menghentikan instance Anda secara paksa.

AWS CLI

Untuk memaksa menghentikan instance menggunakan AWS CLI

Gunakan perintah [stop-instances](#) dan opsi `--force` sebagai berikut:

```
aws ec2 stop-instances --instance-ids i-0123ab456c789d01e --force
```

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](#).

Buat instans pengganti

Untuk mencoba menyelesaikan masalah saat Anda menunggu bantuan dari [AWS re:Post](#) atau [Pusat Dukungan](#), buat instans pengganti. Buat AMI dari instans yang macet, lalu luncurkan instans baru menggunakan AMI yang baru.

Important

Membuat instans pengganti disarankan jika mendaftarkan [pemeriksaan status sistem](#) saja, karena pemeriksaan status instans akan mengakibatkan AMI menyalin replika OS yang rusak. Setelah Anda mengonfirmasi pesan status, buat AMI dan luncurkan instans baru menggunakan AMI yang baru.

Console

Untuk membuat instans pengganti menggunakan konsol

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. Pilih Tidak melakukan boot ulang.
 - c. Pilih Buat gambar.

Untuk informasi selengkapnya, lihat [the section called “Buat AMI dari sebuah instance”](#).

5. Luncurkan instans baru dari AMI dan verifikasi bahwa instans baru tersebut berfungsi.
6. Pilih instans yang macet, lalu pilih Tindakan, Status instans, Akhiri instans. Jika instans tersebut juga mengalami kemacetan saat pengakhiran, Amazon EC2 akan mengakhiri paksa secara otomatis dalam beberapa jam.

AWS CLI

Untuk membuat instans pengganti menggunakan CLI

1. Buat AMI dari instans yang macet menggunakan perintah (AWS CLI) [create-image](#) dan opsi `--no-reboot` sebagai berikut:

```
aws ec2 create-image --instance-id i-0123ab456c789d01e --name "AMI" --  
description "AMI for replacement instance" --no-reboot
```

2. Luncurkan instans baru dari AMI menggunakan perintah (AWS CLI) [run-instances](#) sebagai berikut:

```
aws ec2 run-instances --image-id ami-1a2b3c4d --count 1 --instance-type c3.large  
--key-name MyKeyPair --security-groups MySecurityGroup
```

3. Verifikasi bahwa instans baru berfungsi.
4. Akhiri instans yang macet menggunakan perintah (AWS CLI) [terminate-instances](#) sebagai berikut:

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

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 instans yang macet, lalu pilih Status instans, Akhiri instans. Jika instans tersebut juga mengalami kemacetan saat pengakhiran, Amazon EC2 akan mengakhiri paksa secara otomatis dalam beberapa jam.

Memecahkan masalah penghentian instans (mematikan)

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 instans Anda tetap berada dalam status `shutting-down` selama lebih dari beberapa menit, ada kemungkinan terjadi penundaan karena skrip pematian dijalankan oleh instans tersebut.

Penyebab lain yang mungkin terjadi adalah ada masalah dengan komputer host yang mendasari. Jika instans Anda tetap berada dalam status `shutting-down` selama beberapa jam, Amazon EC2 akan menganggapnya sebagai instans yang macet dan akan menghentikannya secara paksa.

Jika instans Anda macet saat penghentian dan terjadi selama lebih dari beberapa jam, unggah permintaan bantuan ke [re:Post AWS](#). 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](#).

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 `instanceApiTermination` 'nonaktifkan'

Jika Anda mencoba menghentikan instans dan mendapatkan pesan kesalahan `The instance instance_id 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. Untuk informasi selengkapnya, lihat [Aktifkan perlindungan pengakhiran](#).

Anda harus menonaktifkan perlindungan penghentian sebelum Anda dapat menghentikan instans.

Untuk menonaktifkan perlindungan penghentian menggunakan konsol Amazon EC2, pilih instans, lalu pilih Tindakan, Pengaturan Instans, Ubah Perlindungan Penghentian.

Untuk menonaktifkan perlindungan terminasi menggunakan AWS CLI, gunakan perintah berikut.

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-disable-api-termination
```

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, lihat [Panduan Pengguna Amazon EC2 Auto Scaling](#), [Armada EC2](#), atau [Membuat permintaan Armada Spot](#).

Memecahkan masalah instance Linux dengan pemeriksaan status yang gagal

Note

Topik ini hanya berlaku untuk instance Linux.

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 Anda](#).

Daftar Isi

- [Meninjau informasi pemeriksaan status](#)
- [Mengambil log sistem](#)
- [Memecahkan masalah kesalahan log sistem untuk instance Linux](#)
- [Kehabisan memori: hentikan proses](#)
- [KESALAHAN: mmu_update gagal \(Pembaruan manajemen memori gagal\)](#)
- [Kesalahan I/O \(kegagalan perangkat blok\)](#)
- [KESALAHAN I/O: bukan disk lokal maupun jarak jauh \(Perangkat blok terdistribusi yang rusak\)](#)
- [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" \(Modul kernel tidak ada\)](#)
- [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 jumlah perangkat root utama/minor... \(Ketidakcocokan sistem file root/perangkat\)](#)
- [XENBUS: Perangkat tanpa driver...](#)
- [... hari tanpa diperiksa, pemeriksaan paksa \(Diperlukan pemeriksaan sistem file\)](#)
- [fsck mati dengan status keluar... \(Perangkat tidak ada\)](#)
- [Perintah GRUB \(grubdom>\)](#)
- [Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. \(Alamat MAC dengan hard-coding\)](#)
- [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. Di panel detail, pilih Status dan alarm untuk melihat hasil individu untuk semua Pemeriksaan status sistem dan Pemeriksaan status instans.

Jika pemeriksaan status sistem gagal, Anda dapat mencoba salah satu opsi berikut ini:

- Membuat alarm pemulihan instans. Untuk informasi selengkapnya, lihat [Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans](#).
- Jika Anda mengubah jenis instans menjadi [instance yang dibangun di Sistem AWS Nitro](#), pemeriksaan status akan gagal jika Anda bermigrasi dari instance yang tidak memiliki driver ENA dan NVMe yang diperlukan. Untuk informasi selengkapnya, lihat [Kompatibilitas untuk mengubah tipe instans](#).
- Untuk instans yang menggunakan AMI yang didukung Amazon EBS, hentikan dan mulai ulang instans tersebut.
- Untuk instans yang menggunakan AMI yang didukung penyimpanan instans, akhiri instans tersebut dan luncurkan penggantinya.
- Tunggu hingga Amazon EC2 menyelesaikan masalah.

- Buat pos masalah Anda ke [AWS re:Post](#).
- Jika instans Anda berada di grup Auto Scaling, layanan Amazon EC2 Auto Scaling akan secara otomatis meluncurkan instans pengganti. Untuk informasi selengkapnya, lihat [Pemeriksaan Kondisi untuk Instans Penskalaan Otomatis](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- Mengambil log sistem dan mencari kesalahan.

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 instance Linux

Untuk instance Linux yang gagal dalam pemeriksaan status instans, 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

- [Kesalahan I/O \(kegagalan perangkat blok\)](#)
- [KESALAHAN I/O: bukan disk lokal maupun jarak jauh \(Perangkat blok terdistribusi yang rusak\)](#)

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” \(Modul kernel tidak ada\)](#)
- [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 jumlah perangkat root utama/minor... \(Ketidakcocokan sistem file root/perangkat\)](#)
- [XENBUS: Perangkat tanpa driver...](#)
- [... hari tanpa diperiksa, pemeriksaan paksa \(Diperlukan pemeriksaan sistem file\)](#)
- [fsck mati dengan status keluar... \(Perangkat tidak ada\)](#)

Kesalahan Sistem Operasi

- [Perintah GRUB \(grubdom>\)](#)
- [Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. \(Alamat MAC dengan hard-coding\)](#)
- [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

out-of-memory Kesalahan ditunjukkan 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
Didukung Amazon EBS	<p>Lakukan salah satu dari berikut ini:</p> <ul style="list-style-type: none"> • 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. • Boot ulang instans untuk mengembalikannya ke status tidak terganggu. Masalah mungkin akan terjadi lagi kecuali Anda mengubah tipe instans.
Didukung penyimpanan instans	<p>Lakukan salah satu dari berikut ini:</p> <ul style="list-style-type: none"> • Akhiri instans dan luncurkan instans baru, dengan menentukan tipe instans yang berbeda. Misalnya, tipe instans yang lebih besar atau instans memori yang dioptimalkan.

Untuk tipe instans ini	Lakukan hal berikut
	<ul style="list-style-type: none">• 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

Buat pos masalah Anda ke [Forum Developer](#) atau hubungi [AWS Support](#).

Kesalahan I/O (kegagalan perangkat blok)

Kesalahan input/output diindikasikan dengan entri log sistem yang serupa 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
Didukung Amazon EBS	Volume Amazon EBS yang gagal
Didukung penyimpanan instans	Drive fisik yang gagal

Tindakan yang disarankan

Untuk tipe instans ini	Lakukan hal berikut
Didukung Amazon EBS	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Hentikan instans. 2. Lepaskan lampiran volume. 3. Coba pulihkan volume.

Untuk tipe instans ini	Lakukan hal berikut
	<p data-bbox="898 247 1016 279"> Note</p> <p data-bbox="946 304 1474 527">Disarankan untuk sering membuat snapshot volume Amazon EBS. Hal ini secara signifikan mengurangi risiko kehilangan data sebagai akibat dari kegagalan.</p> <ol data-bbox="829 590 1409 674" style="list-style-type: none">4. Lampirkan kembali volume ke instans.5. Mulai instans.
Didukung penyimpanan instans	<p data-bbox="829 758 1419 789">Akhiri instans dan luncurkan instans baru.</p> <p data-bbox="862 873 980 905"> Note</p> <p data-bbox="911 930 1438 1014">Data tidak dapat dipulihkan. Pulihkan dari cadangan.</p> <p data-bbox="862 1161 980 1192"> Note</p> <p data-bbox="911 1218 1438 1493">Disarankan agar menggunakan Amazon S3 atau Amazon EBS untuk pencadangan. Volume penyimpanan instans secara langsung terikat dengan kegagalan host tunggal dan disk tunggal.</p>

KESALAHAN I/O: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi yang rusak)

Kesalahan input/output pada perangkat diindikasikan dengan entri log sistem yang serupa 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
Didukung Amazon EBS	Volume Amazon EBS yang gagal
Didukung penyimpanan instans	Drive fisik yang gagal

Tindakan yang disarankan

Akhiri instans dan luncurkan instans baru.

Untuk instans yang didukung Amazon EBS, 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
```

Tindakan yang disarankan

Untuk tipe instans ini	Lakukan hal berikut
Didukung Amazon EBS	<p>Gunakan kernel yang lebih baru, yang berbasis GRUB ataupun statis, menggunakan salah satu opsi berikut:</p> <p>Opsi 1: Akhiri instans dan luncurkan instans baru, dengan menentukan parameter <code>-kernel</code> dan <code>-ramdisk</code>.</p> <p>Opsi 2:</p> <ol style="list-style-type: none"> 1. Hentikan instans. 2. Modifikasi atribut kernel dan ramdisk untuk menggunakan kernel yang lebih baru. 3. Mulai instans.
Didukung penyimpanan instans	Akhiri instans dan luncurkan instans baru, dengan menentukan parameter <code>-kernel</code> dan <code>-ramdisk</code> .

“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
Didukung Amazon EBS	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Hentikan instans. 2. Modifikasi konfigurasi untuk menggunakan kernel yang lebih baru. 3. Mulai instans.
Didukung penyimpanan instans	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Buat AMI yang menggunakan kernel yang lebih baru. 2. Akhiri instans. 3. Mulai instans baru dari AMI yang Anda buat.

“FATAL: Tidak dapat memuat /lib/modules” atau “BusyBox” (Modul kernel tidak ada)

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
Didukung Amazon EBS	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Pilih ramdisk yang sudah benar untuk volume Amazon EBS. 2. Hentikan instans. 3. Lepaskan lampiran volume dan perbaiki. 4. Lampirkan volume ke instans. 5. Mulai instans. 6. Modifikasi AMI untuk menggunakan ramdisk yang telah diperbaiki.
Didukung penyimpanan instans	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Akhiri instans dan luncurkan instans baru dengan ramdisk yang benar. 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
Didukung Amazon EBS	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Hentikan instans. 2. Ganti dengan kernel yang berfungsi. 3. Instal kernel pengganti. 4. Modifikasi AMI dengan memperbaiki kernel.
Didukung penyimpanan instans	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Akhiri instans dan luncurkan instans baru dengan kernel yang benar.

Untuk tipe instans ini	Lakukan hal berikut
	<ol style="list-style-type: none">2. Buat AMI baru dengan kernel yang benar.3. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan AWS Support.

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

- Terdapat bug dalam definisi sistem file ramdisk `/etc/fstab`
- Definisi sistem file salah konfigurasi di `/etc/fstab`
- Drive tidak ada/gagal

Tindakan yang disarankan

Untuk tipe instans ini	Lakukan hal berikut
Didukung Amazon EBS	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Hentikan instans, lepaskan lampiran volume root, perbaiki/modifikasi <code>/etc/fstab</code> volume, lampirkan volume ke instans, dan mulai instans. 2. Perbaiki ramdisk untuk menyertakan <code>/etc/fstab</code> yang dimodifikasi (jika berlaku). 3. Modifikasi AMI untuk menggunakan ramdisk yang lebih baru. <p>Bidang keenam di <code>fstab</code> menentukan persyaratan ketersediaan pemasangan –nilai bukan nol mengimplikasikan bahwa <code>fsck</code> 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-</p>

Untuk tipe instans ini	Lakukan hal berikut
	hatilah dengan fitur ini dan baca halaman manual Linux untuk fstab.
Didukung penyimpanan instans	Gunakan prosedur berikut: <ol style="list-style-type: none">1. Akhiri instans dan luncurkan instans baru.2. Lepaskan lampiran volume Amazon EBS yang tidak sesuai dan boot ulang instans.3. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan AWS Support.

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
Didukung Amazon EBS	<ul style="list-style-type: none"> • Volume Amazon EBS yang terlepas lampirannya atau gagal. • Sistem file rusak. • Kombinasi ramdisk dan AMI yang tidak cocok (seperti ramdisk Debian dengan SUSE AMI).
Didukung penyimpanan instans	<ul style="list-style-type: none"> • Drive yang gagal. • Sistem file yang rusak. • Ramdisk dan kombinasi yang tidak cocok (misalnya ramdisk Debian dengan SUSE AMI).

Tindakan yang disarankan

Untuk tipe instans ini	Lakukan hal berikut
Didukung Amazon EBS	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Hentikan instans.

Untuk tipe instans ini	Lakukan hal berikut
	<ol style="list-style-type: none">2. Lepaskan lampiran volume root.3. Lampirkan volume root ke instans yang berfungsi yang diketahui.4. Jalankan pemeriksaan sistem file (<code>fsck -a / dev/...</code>).5. Perbaiki kesalahan apa pun.6. Lepaskan lampiran volume dari instans yang berfungsi yang diketahui.7. Lampirkan volume ke instans yang dihentikan.8. Mulai instans.9. Periksa ulang status instans.
Didukung penyimpanan instans	<p>Cobalah salah satu cara berikut ini:</p> <ul style="list-style-type: none">• Mulai instans baru.• (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan AWS Support.

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
Didukung Amazon EBS	<ul style="list-style-type: none"> • Perangkat tidak terlampir dengan benar. • Perangkat root tidak terpasang pada titik perangkat yang benar. • Sistem file tidak dalam format yang diharapkan. • Penggunaan kernel warisan (seperti 2.6.16-XenU). • 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
Didukung Amazon EBS	<p>Lakukan salah satu dari berikut ini:</p> <ul style="list-style-type: none"> • Hentikan lalu mulai ulang instans. • Modifikasi volume root untuk dilampirkan pada titik perangkat yang benar, kemungkinan <code>/dev/sda1</code> alih-alih <code>/dev/sda</code>. • Hentikan dan modifikasi untuk menggunakan kernel modern. • 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 jumlah perangkat root utama/minor... (Ketidakcocokan sistem file root/perangkat)

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>udev[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
Didukung Amazon EBS	Gunakan prosedur berikut: <ol style="list-style-type: none"> 1. Hentikan instans. 2. Lepaskan lampiran volume. 3. Perbaiki masalah pemetaan perangkat.

Untuk tipe instans ini	Lakukan hal berikut
	<ol style="list-style-type: none"> 4. Mulai instans. 5. Modifikasi AMI untuk menangani masalah pemetaan perangkat.
Didukung penyimpanan instans	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Buat AMI baru dengan perbaikan yang sesuai (petakan perangkat blok dengan benar). 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>udev[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
Didukung Amazon EBS	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Hentikan instans. 2. Lepaskan lampiran volume. 3. Perbaiki masalah pemetaan perangkat. 4. Mulai instans. 5. Modifikasi AMI untuk menangani masalah pemetaan perangkat.
Didukung penyimpanan instans	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Buat AMI dengan perbaikan yang sesuai (petakan perangkat blok dengan benar). 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
Didukung Amazon EBS	<p>Coba salah satu atau beberapa cara berikut untuk menyelesaikan masalah:</p> <ul style="list-style-type: none"> • Hentikan instans, lampirkan volume ke instans yang sudah ada yang sedang berjalan.

Untuk tipe instans ini	Lakukan hal berikut
	<ul style="list-style-type: none">• Jalankan pemeriksaan konsistensi secara manual.• Perbaiki ramdisk untuk menyertakan utilitas yang relevan.• Modifikasi parameter penyetelan sistem file untuk menghapus persyaratan konsistensi (tidak disarankan).
Didukung penyimpanan instans	<p>Coba salah satu atau beberapa cara berikut untuk menyelesaikan masalah:</p> <ul style="list-style-type: none">• Buat ulang paket ramdisk dengan alat yang benar.• Modifikasi parameter penyetelan sistem file untuk menghapus persyaratan konsistensi (tidak disarankan).• Akhiri instans dan luncurkan instans baru.• (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan AWS Support.

Perintah GRUB (grubdom>)

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
Didukung Amazon EBS	<ul style="list-style-type: none"> • File konfigurasi GRUB tidak ada. • Citra GRUB yang digunakan salah, mengharapkan file konfigurasi GRUB berada di lokasi yang berbeda. • 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	<ul style="list-style-type: none"> • File konfigurasi GRUB tidak ada. • Citra GRUB yang digunakan salah, mengharapkan file konfigurasi GRUB berada di lokasi yang berbeda. • 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
Didukung Amazon EBS	<p>Opsi 1: Modifikasi AMI dan luncurkan ulang instans:</p> <ol style="list-style-type: none"> 1. Modifikasi AMI sumber untuk membuat file konfigurasi GRUB di lokasi standar (<code>/boot/grub/menu.lst</code>).

Untuk tipe instans ini	Lakukan hal berikut
	<ol style="list-style-type: none">2. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasarinya dan tingkatkan GRUB jika perlu.3. Pilih citra GRUB yang sesuai, (drive hd0-1st atau hd00 – drive pertama, partisi pertama).4. Akhiri instans dan luncurkan instans baru menggunakan AMI yang Anda buat. <p>Opsi 2: Perbaiki instans yang sudah ada:</p> <ol style="list-style-type: none">1. Hentikan instans.2. Lepaskan lampiran sistem file root.3. Lampirkan sistem file root ke instans yang berfungsi yang diketahui.4. Pasang sistem file.5. Buat file konfigurasi GRUB6. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasarinya dan tingkatkan GRUB jika perlu.7. Lepaskan lampiran sistem file.8. Lampirkan ke instans asli.9. Modifikasi atribut kernel untuk menggunakan citra GRUB yang sesuai (disk pertama atau partisi pertama pada disk pertama).10. Mulai instans.

Untuk tipe instans ini	Lakukan hal berikut
Didukung penyimpanan instans	<p>Opsi 1: Modifikasi AMI dan luncurkan ulang instans:</p> <ol style="list-style-type: none">1. Buat AMI baru dengan file konfigurasi GRUB di lokasi standar (/boot/grub/menu.lst).2. Pilih citra GRUB yang sesuai, (drive hd0-1st atau hd00 – drive pertama, partisi pertama).3. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasarinya dan tingkatkan GRUB jika perlu.4. Akhiri instans dan luncurkan instans baru menggunakan AMI yang telah Anda buat. <p>Opsi 2: Akhiri instans dan luncurkan instans baru, dengan menentukan kernel yang benar.</p> <div data-bbox="829 999 1507 1266" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> Note</p><p>Untuk memulihkan data dari instans yang sudah ada, hubungi AWS Support.</p></div>

Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat MAC dengan hard-coding)

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
Didukung Amazon EBS	<p>Lakukan salah satu dari berikut ini:</p> <ul style="list-style-type: none"> • Modifikasi AMI untuk menghapus hard-coding dan meluncurkan ulang instans. • Modifikasi instans untuk menghapus alamat MAC dengan hard-coding. <p>ATAU</p> <p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Hentikan instans. 2. Lepaskan lampiran volume root. 3. Lampirkan volume ke instans lain dan modifikasi volume untuk menghapus alamat MAC dengan hard-coding. 4. Lampirkan volume ke instans asli. 5. Mulai instans.
Didukung penyimpanan instans	<p>Lakukan salah satu dari berikut ini:</p> <ul style="list-style-type: none"> • Modifikasi instans untuk menghapus alamat MAC dengan hard-coding. • 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
Didukung Amazon EBS	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none">1. Hentikan instans yang gagal.2. Lepaskan lampiran volume root instans yang gagal.3. Lampirkan volume root ke instans Linux lain yang sedang berjalan (yang nantinya akan disebut sebagai instans pemulihan).4. Hubungkan ke instans pemulihan dan pasang volume root dari instans yang gagal.5. Nonaktifkan SELinux pada volume root yang dipasang. Proses ini bervariasi di seluruh distribusi Linux. Untuk informasi selengkapnya, lihat dokumentasi khusus OS Anda.

Untuk tipe instans ini	Lakukan hal berikut
	<div data-bbox="867 212 1507 716" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-bottom: 10px;"> <p>Note</p> <p>Pada beberapa sistem, Anda menonaktifkan SELinux dengan mengatur SELINUX=disabled dalam file <code>/mount_point /etc/sysconfig/selinux</code> , dengan <code>mount_point</code> adalah lokasi tempat Anda memasang volume pada instans pemulihan.</p> </div> <ol style="list-style-type: none"> 6. Copot dan lepaskan lampiran volume root dari instans pemulihan dan lampirkan kembali ke instans asli. 7. Mulai instans.
Didukung penyimpanan instans	<p>Gunakan prosedur berikut:</p> <ol style="list-style-type: none"> 1. Akhiri instans dan luncurkan instans baru. 2. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan AWS Support.

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
Didukung Amazon EBS	Lakukan salah satu dari berikut ini: <ul style="list-style-type: none">• Modifikasi AMI dan instans untuk menggunakan kernel modern dan luncurkan ulang instans.• Boot ulang instans.
Didukung penyimpanan instans	Lakukan salah satu dari berikut ini: <ul style="list-style-type: none">• Akhiri instans.• Modifikasi AMI untuk menggunakan kernel modern, dan luncurkan instans baru menggunakan AMI ini.

Memecahkan masalah booting instance Linux dari volume yang salah

Note

Topik pemecahan masalah ini hanya berlaku untuk instance Linux.

Dalam beberapa situasi, Anda mungkin menemukan volume selain volume yang terpasang pada /dev/xvda atau /dev/sda telah menjadi volume root dari instans Anda. 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 dapat menjadikan `/dev/xvdf1` sebagai perangkat root yang menjadi tujuan boot instans Anda setelah ramdisk awal dijalankan, alih-alih volume `/dev/xvda1` yang Anda inginkan untuk melakukan 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 Sysprep dengan instance Windows

Note

Topik pemecahan masalah ini hanya berlaku untuk instance 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, EC2Launch 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 Log Terkait Penataan 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.

Penggunaan EC2Rescue untuk Linux

EC2Rescue untuk Linux adalah alat sumber terbuka yang dapat dijalankan pada instans Amazon EC2 Linux untuk mendiagnosis dan memecahkan masalah umum menggunakan pustaka lebih dari 100 modul. `easy-to-use` Beberapa kasus penggunaan umum EC2Rescue untuk Linux termasuk mengumpulkan log syslog dan log pengelola paket, mengumpulkan data pemanfaatan sumber daya, dan mendiagnosis/memulihkan parameter kernel bermasalah yang diketahui dan masalah umum OpenSSH.

`AWS Support-TroubleshootSSH` runbook menginstal EC2Rescue untuk Linux dan kemudian menggunakan alat tersebut 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 [AWS Support-TroubleshootSSH](#).

Jika Anda menggunakan instance Windows, lihat [the section called “EC2Rescue untuk Windows Server”](#).

Daftar Isi

- [Menginstal EC2Rescue untuk Linux](#)
- [Bekerja dengan EC2Rescue untuk Linux](#)
- [Mengembangkan modul EC2Rescue](#)

Menginstal EC2Rescue untuk 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+

AWSSupport-TroubleshootSSH runbook menginstal EC2Rescue untuk Linux dan kemudian menggunakan alat tersebut 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 [AWS Support-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/ec2r1.tgz
```

2. (Opsional) Sebelum melanjutkan, Anda dapat memverifikasi tanda tangan file instalasi EC2Rescue untuk Linux secara opsional. Untuk informasi selengkapnya, lihat [\(Opsional\) Verifikasi tanda tangan EC2Resuce untuk Linux](#).

3. Unduh file sha256 hash:

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2r1.tgz.sha256
```

4. Verifikasi integritas tarball:

```
sha256sum -c ec2r1.tgz.sha256
```

5. Buka kemasan tarball:

```
tar -xzf ec2r1.tgz
```

6. Verifikasi instalasi dengan mencantumkan file bantuan:

```
cd ec2r1-<version_number>
./ec2r1 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 direkomendasikan untuk memverifikasi validitas paket EC2Rescue untuk Linux untuk sistem operasi berbasis Linux.

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.

File EC2Rescue untuk Linux untuk sistem operasi berbasis Linux ditandatangani dengan menggunakan GnuPG, yaitu 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 lebih lanjut tentang PGP dan GnuPG (GPG), lihat <http://www.gnu.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

- [Menginstal alat GPG](#)
- [Autentikasi dan impor kunci publik](#)
- [Memverifikasi tanda tangan paket](#)

Menginstal alat GPG

Jika sistem operasi Anda adalah Linux atau Unix, alat GPG mungkin sudah terinstal. Untuk menguji apakah alat ini sudah terinstal di sistem Anda, masukkan `gpg2` pada prompt perintah. Jika alat GPG telah terinstal, Anda akan melihat prompt perintah GPG. Jika alat GPG belum terinstal, Anda akan melihat pesan kesalahan yang menyatakan bahwa perintah tidak dapat ditemukan. Anda dapat menginstal paket GnuPG dari repositori.

Untuk menginstal alat GPG pada Linux berbasis Debian

- Dari terminal, jalankan perintah berikut:

```
apt-get install gnupg2
```

Untuk menginstal alat GPG pada Linux berbasis Red Hat

- Dari terminal, jalankan perintah berikut:

```
yum install gnupg2
```

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/ec2r1.key
```

2. Pada prompt perintah di direktori tempat Anda menyimpan `ec2r1.key`, gunakan perintah berikut untuk mengimpor kunci publik EC2Rescue untuk Linux ke dalam keyring Anda:

```
gpg2 --import ec2r1.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)
```

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/ec2rescuelineux/ec2r1.tgz.sig
```

2. Verifikasi tanda tangan dengan menjalankan perintah berikut pada prompt perintah di direktori tempat Anda menyimpan `ec2r1.tgz.sig` dan file instalasi EC2Rescue untuk Linux. Kedua file tersebut harus ada.

```
gpg2 --verify ./ec2r1.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 lebih lanjut, lihat <http://www.gnupg.org>.

Bekerja dengan EC2Rescue untuk Linux

Berikut ini adalah tugas umum yang dapat Anda lakukan untuk mulai menggunakan alat ini.

Tugas

- [Menjalankan EC2Rescue untuk Linux](#)
- [Unggah hasilnya](#)
- [Membuat cadangan](#)
- [Mencari bantuan](#)

Menjalankan EC2Rescue untuk Linux

Anda dapat menjalankan EC2Rescue untuk Linux seperti yang ditunjukkan dalam contoh berikut.

Example Contoh: Jalankan semua modul

Untuk menjalankan semua modul, jalankan EC2Rescue untuk Linux, tanpa opsi:

```
./ec2r1 run
```

Beberapa modul memerlukan akses root. Jika Anda bukan pengguna root, gunakan sudo untuk menjalankan modul ini sebagai berikut:

```
sudo ./ec2r1 run
```

Example Contoh: Jalankan modul tertentu

Untuk menjalankan modul tertentu saja, gunakan parameter `--only-modules`:

```
./ec2r1 run --only-modules=module_name --arguments
```

Misalnya, perintah ini menjalankan modul dig untuk mencari domain `amazon.com`:

```
./ec2r1 run --only-modules=dig --domain=amazon.com
```

Example Contoh: Lihat hasilnya

Anda dapat melihat hasilnya di `/var/tmp/ec2r1`:

```
cat /var/tmp/ec2r1/logfile_location
```

Misalnya, lihat file log untuk modul dig:

```
cat /var/tmp/ec2r1/2017-05-11T15_39_21.893145/mod_out/run/dig.log
```

Unggah hasilnya

Jika AWS Support telah meminta hasil atau untuk membagikan hasil dari bucket S3, unggah menggunakan alat CLI EC2Rescue for Linux. Output dari perintah EC2Rescue untuk Linux harus memberikan perintah yang perlu Anda gunakan.

Example Contoh: Unggah hasil ke AWS Support

```
./ec2r1 upload --upload-directory=/var/tmp/ec2r1/2017-05-11T15_39_21.893145 --support-url="URLProvidedByAWSsupport"
```

Example Contoh: Mengunggah hasil ke bucket S3

```
./ec2r1 upload --upload-directory=/var/tmp/ec2r1/2017-05-11T15_39_21.893145 --presigned-url="YourPresignedS3URL"
```

Untuk informasi lebih lanjut tentang cara membuat URL pre-signed untuk Amazon S3, lihat [Mengunggah Objek Menggunakan URL Pre-Signed](#).

Membuat cadangan

Buatlah cadangan untuk instans Anda, satu volume atau lebih, atau ID perangkat tertentu dengan menggunakan perintah berikut.

Example Contoh: Mencadangkan instans dengan menggunakan Amazon Machine Image (AMI)

```
./ec2r1 run --backup=ami
```

Example Contoh: Mencadangkan semua volume yang terkait dengan instans

```
./ec2r1 run --backup=allvolumes
```

Example Contoh: Mencadangkan volume tertentu

```
./ec2r1 run --backup=volumeID
```

Mencari bantuan

EC2Rescue untuk Linux menyertakan file bantuan yang memberi Anda informasi dan sintaks untuk setiap perintah yang tersedia.

Example Contoh: Menampilkan bantuan umum

```
./ec2r1 help
```

Example Contoh: Membuat daftar modul yang tersedia

```
./ec2r1 list
```

Example Contoh: Menampilkan bantuan untuk modul tertentu

```
./ec2r1 help module_name
```

Misalnya, gunakan perintah berikut untuk menunjukkan file bantuan untuk modul dig:

```
./ec2r1 help dig
```

Mengembangkan modul EC2Rescue

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.
title	Judul singkat dan deskriptif untuk modul. Panjang judul harus berisi 50 karakter atau kurang.
helptext	<p>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.</p> <p>Misalnya:</p> <pre>helptext: !!str Collect output from ps for system analysis Consumes --times= for number of times to repeat Consumes --period= for time period between repetition</pre>
placement	<p>Tahap ketika modul harus dijalankan. Nilai yang didukung:</p> <ul style="list-style-type: none">• prediagnostic

Atribut	Deskripsi
	<ul style="list-style-type: none">• run• postdiagnostic
language	<p>Bahasa yang digunakan untuk menuliskan kode modul. Nilai yang didukung:</p> <ul style="list-style-type: none">• bash• python <div data-bbox="829 642 1507 863" style="border: 1px solid #ccc; border-radius: 10px; padding: 10px;"><p> Note</p><p>Kode Python harus kompatibel dengan Python 2.7.9+ maupun Python 3.2+.</p></div>
remediation	<p>Menunjukkan apakah modul mendukung remediasi. Nilai yang didukung adalah True atau False.</p> <p>Secara default, modul menjadi False jika hal ini tidak ada, sehingga membuatnya menjadi atribut opsional untuk modul yang tidak mendukung remediasi.</p>
content	Keseluruhan kode skrip.
constraint	Nama objek yang berisi nilai kendala.
domain	<p>Penjelasan tentang cara modul dikelompokkan atau diklasifikasikan. Rangkaian modul yang disertakan menggunakan domain berikut:</p> <ul style="list-style-type: none">• application• net• os• performance

Atribut	Deskripsi
class	<p>Penjelasan tentang tipe tugas yang dilakukan oleh modul. Rangkaian modul yang disertakan menggunakan domain berikut:</p> <ul style="list-style-type: none">• collect (mengambil output dari program)• diagnose (lulus/gagal berdasarkan serangkaian kriteria)• gather (menyalin file dan menulis ke file tertentu)
distro	<p>Daftar distribusi Linux yang didukung oleh modul ini. Rangkaian modul yang disertakan menggunakan distribusi berikut:</p> <ul style="list-style-type: none">• alami (Amazon Linux)• rhel• ubuntu• suse
diperlukan	<p>Argumen wajib yang digunakan oleh modul dari opsi CLI.</p>
optional	<p>Argumen opsional yang dapat digunakan oleh modul.</p>
software	<p>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 menjalankan modul.</p>

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	<p>Mengindikasikan apakah akses root diperlukan untuk menjalankan modul.</p> <p>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.</p>
perfimpact	Mengindikasikan apakah modul dapat menimbulkan dampak performa yang signifikan terhadap lingkungan tempat modul dijalankan. Jika nilainya benar dan argumen <code>--perfimpact=true</code> 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 <code>ec2rl.py</code> . Path ini dapat digunakan untuk menemukan direktori lib dan menggunakan modul Python yang bervendor.
EC2RL_WORKDIR	Direktori tmp utama untuk alat diagnostik. Nilai default: <code>/var/tmp/ec2rl</code> .
EC2RL_RUNDIR	Direktori tempat semua output disimpan. Nilai default: <code>/var/tmp/ec2rl/<date&timestamp></code> .
EC2RL_GATHEREDDIR	Direktori root untuk menempatkan data modul yang dikumpulkan. Nilai default: <code>/var/tmp/ec2rl/<date&timestamp>/mod_out/gathered/</code> .
EC2RL_NET_DRIVER	Driver yang digunakan untuk antarmuka jaringan non-virtual pertama yang diurutkan menurut abjad pada instans. Contoh: <ul style="list-style-type: none">• <code>xen_netfront</code>• <code>ixgbevf</code>• <code>ena</code>
EC2RL_SUDO	Betul jika EC2Rescue untuk Linux dijalankan sebagai root; jika tidak, salah.
EC2RL_VIRT_TYPE	Tipe virtualisasi seperti yang disediakan oleh metadata instans. Contoh:

Variabel Lingkungan	Deskripsi
	<ul style="list-style-type: none"> • default-hvm • default-paravirtual
EC2RL_INTERFACES	Daftar antarmuka yang disebutkan pada sistem. Nilai tersebut adalah string yang berisi nama, seperti eth0, eth1, dan lain-lain. Ini dibuat melalui <code>functions.bash</code> dan hanya tersedia untuk modul yang bersumber darinya.

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
```

Gunakan EC2Rescue untuk Windows Server

EC2Rescue untuk Windows Server adalah easy-to-use alat yang Anda jalankan pada instans Amazon EC2 Windows Server 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.

EC2Rescue untuk Windows Server memiliki dua modul yang berbeda: modul pengumpul data yang mengumpulkan data dari semua sumber yang berbeda, dan modul penganalisis yang menguraikan data yang dikumpulkan terhadap serangkaian aturan yang telah ditetapkan untuk mengidentifikasi masalah serta 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.

Runbook `AWSSupport-ExecuteEC2Rescue` menggunakan alat EC2Rescue untuk memecahkan masalah dan, jika mungkin, memperbaiki masalah konektivitas umum dengan instans EC2 tertentu. Untuk informasi selengkapnya, dan untuk menjalankan otomatisasi ini, lihat [AWSSupport-ExecuteEC2Rescue](#).

Jika Anda menggunakan instance Linux, lihat [the section called “EC2Rescue untuk Linux”](#).

Daftar Isi

- [Gunakan GUI EC2Rescue untuk Windows Server](#)
- [Gunakan EC2Rescue untuk Windows Server dengan baris perintah](#)
- [Gunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command](#)

Gunakan GUI EC2Rescue untuk Windows Server

EC2Rescue untuk Windows Server dapat melakukan analisis berikut pada instans offline:

Opsi	Deskripsi
Diagnosis dan Penyelamatan	<p>EC2Rescue untuk Windows Server dapat mendeteksi dan mengatasi masalah dengan pengaturan layanan berikut ini:</p> <ul style="list-style-type: none"> • Waktu Sistem

Opsi	Deskripsi
	<ul style="list-style-type: none">• RealTimeisUniversal - Mendeteksi apakah kunci RealTimeisUniversal registri diaktifkan. Jika dinonaktifkan, waktu sistem Windows menyimpang saat zona waktu disetel ke nilai selain UTC. • Windows Firewall<ul style="list-style-type: none">• Jaringan domain - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.• Jaringan pribadi - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.• Jaringan tamu atau publik - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan. • Desktop Jarak Jauh<ul style="list-style-type: none">• Mulai Layanan - Mendeteksi apakah layanan Desktop Jarak Jauh diaktifkan.• Koneksi Desktop Jarak Jauh - Mendeteksi apakah fitur ini diaktifkan.• Port TCP - Mendeteksi port mana yang mendengarkan layanan Desktop Jarak Jauh. • EC2Config (Windows Server 2012 R2 dan versi sebelumnya)<ul style="list-style-type: none">• Instalasi - Mendeteksi versi EC2Config apa yang diinstal.

Opsi	Deskripsi
	<ul style="list-style-type: none">• Mulai Layanan - Mendeteksi apakah layanan EC2Config diaktifkan atau dinonaktifkan.• Ec2 SetPassword - Menghasilkan kata sandi administrator baru.• HandleUserData Ec2 - Memungkinkan Anda menjalankan skrip data pengguna pada boot berikutnya dari instance. • EC2Launch (Windows Server 2016 dan versi terbarunya)<ul style="list-style-type: none">• Instalasi - Mendeteksi versi EC2Launch apa yang diinstal.• Ec2 SetPassword - Menghasilkan kata sandi administrator baru. • Antarmuka Jaringan<ul style="list-style-type: none">• Startup Layanan DHCP - Mendeteksi apakah layanan DHCP diaktifkan atau dinonaktifkan.• Detail ethernet - Menampilkan informasi tentang versi driver jaringan, jika terdeteksi.• DHCP di Ethernet - Mendeteksi apakah DHCP diaktifkan atau dinonaktifkan. • Status tanda tangan disk<ul style="list-style-type: none">• 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

Opsi	Deskripsi
	menimpa tanda tangan disk dengan tanda tangan pada BCD.
Pulihkan	<p>Lakukan salah satu tindakan berikut:</p> <ul style="list-style-type: none"> Konfigurasi Baik yang Terakhir Diketahui - Berupaya melakukan boot instans ke status terakhir yang diketahui dapat melakukan boot. Pulihkan registri dari cadangan - Memulihkan registri dari <code>\Windows\System32\config\RegBack</code> .
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.

 **Note**

Di Windows Server 2016 dan versi yang lebih baru, log dikumpulkan dalam format Pelacakan Peristiwa untuk Windows (ETW).

Item	Deskripsi
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.
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.

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):

```
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):

```
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.

Gunakan EC2Rescue untuk Windows Server dengan baris perintah

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	<ul style="list-style-type: none"> 'Application' 'System' 'EC2ConfigService' 	Mengumpulkan log aplikasi, sistem, dan peristiwa EC2Config.
memory-dump	<ul style="list-style-type: none"> 'Memory Dump File' 'Mini Dump Files' 	Mengumpulkan file pembuangan memori yang ada pada instans.
ec2config	<ul style="list-style-type: none"> 'Log Files' 'Configuration Files' 	Mengumpulkan file log yang dihasilkan oleh layanan EC2Config.
ec2launch	<ul style="list-style-type: none"> 'Logs' 'Config' 	Mengumpulkan file log yang dihasilkan oleh penulisan EC2Launch.
ssm-agent	<ul style="list-style-type: none"> 'Log Files' 'Patch Baseline Logs' '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	<ul style="list-style-type: none"> 'SetupAPI Log Files' 'DPIInst Log File' 'AWS PV Setup Log File' 	Mengumpulkan log Windows SetupAPI (setupapi.dev.log dan setupapi.setup.log).
registry	<ul style="list-style-type: none"> 'SYSTEM' 'SOFTWARE' 	Mengumpulkan hive SYSTEM dan SOFTWARE.

Grup log	Log yang tersedia	Deskripsi
	<ul style="list-style-type: none"> 'BCD' 	
egpu	<ul style="list-style-type: none"> 'Event Log' 'System Files' 	Mengumpulkan log peristiwa terkait GPU elastis.
boot-config	'BCDEDIT Output'	Mengumpulkan hive HKEY_LOCAL_MACHINE \BCD00000000 .
windows-update	'Log Files'	Mengumpulkan file log yang dihasilkan oleh Windows Update.
		<div style="border: 1px solid #00aaff; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>Di Windows Server 2016 dan lebih baru, log dikumpulkan dalam format Pelacakan Peristiwa untuk Windows (ETW).</p> </div>
cloudendure	<ul style="list-style-type: none"> 'Migrate Script Logs' 'Driver Logs' '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.

Grup log	Log yang tersedia	Deskripsi
gpresult	'GPResult Output'	Mengumpulkan laporan Kebijakan Grup.

Berikut ini adalah opsi yang tersedia:

- `/output: FilePath<output>` - Lokasi jalur file tujuan yang diperlukan 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'	<p>Waktu Sistem</p> <ul style="list-style-type: none"> RealTimeisUniversal - Mendeteksi apakah kunci RealTimeisUniversal registri diaktifkan. Jika dinonaktifkan, waktu sistem Windows menyimpang saat zona waktu disetel ke nilai selain UTC.
firewall	<ul style="list-style-type: none"> 'Domain networks' 'Private networks' 'Guest or public networks' 	<p>Windows Firewall</p> <ul style="list-style-type: none"> Jaringan domain - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan. Jaringan pribadi - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan. Jaringan tamu atau publik - Mendeteksi apakah

Grup layanan	Tindakan yang tersedia	Deskripsi
		profil Windows Firewall ini diaktifkan atau dinonaktifkan.
rdp	<ul style="list-style-type: none"> 'Service Start' 'Remote Desktop Connections' 'TCP Port' 	<p>Desktop Jarak Jauh</p> <ul style="list-style-type: none"> Mulai Layanan - Mendeteksi apakah layanan Desktop Jarak Jauh diaktifkan. Koneksi Desktop Jarak Jauh - Mendeteksi apakah fitur ini diaktifkan. Port TCP - Mendeteksi port mana yang mendengarkan layanan Desktop Jarak Jauh.
ec2config	<ul style="list-style-type: none"> 'Service Start' 'Ec2SetPassword' 'Ec2HandleUserData' 	<p>EC2Config</p> <ul style="list-style-type: none"> Mulai Layanan - Mendeteksi apakah layanan EC2Config diaktifkan. Ec2 SetPassword - Menghasilkan kata sandi administrator baru. HandleUserData Ec2 - Memungkinkan Anda menjalankan skrip data pengguna pada boot berikutnya dari instance.
ec2launch	'Reset Administrator Password'	Buat kata sandi administrator Windows baru.

Grup layanan	Tindakan yang tersedia	Deskripsi
network	'DHCP Service Startup'	Antarmuka Jaringan <ul style="list-style-type: none"> 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 mengoreksi tanda tangan setelah proses offline selesai secara default, bahkan ketika Anda menggunakan opsi `/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
```

Gunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command

AWS Support menyediakan dokumen Systems Manager Run Command untuk berinteraksi dengan instans Systems Manager-enabled 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 ini.
- Menjalankan EC2 RescueCmd 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 harus diteruskan untuk perintah yang ditentukan.

Note

Agar `ResetAccess` tindakan berfungsi, instans Amazon EC2 Anda harus memiliki kebijakan berikut yang dilampirkan untuk menulis kata sandi terenkripsi ke Parameter Store. Harap tunggu beberapa menit sebelum mencoba mengatur ulang kata sandi instans setelah Anda melampirkan kebijakan ini ke peran IAM yang terkait.

Menggunakan kunci KMS default:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
        "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
        Passwords/<instanceid>"
      ]
    }
  ]
}
```

Menggunakan kunci KMS kustom:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
```

```
    "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
    Passwords/<instanceid>"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "kms:Encrypt"
  ],
  "Resource": [
    "arn:aws:kms:region:account_id:key/<kmskeyid>"
  ]
}
]
```

Prosedur berikut menjelaskan cara menampilkan file JSON untuk dokumen ini di konsol Amazon EC2.

Untuk menampilkan JSON pada dokumen Systems Manager Run Command

1. Buka konsol Systems Manager di <https://console.aws.amazon.com/systems-manager/home>.
2. Pada panel navigasi, perbesar Layanan Bersama dan pilih Dokumen.
3. Di bar pencarian, atur Pemilik sebagai Dimiliki oleh Saya atau Amazon dan atur Prefiks nama dokumen ke `AWSSupport-RunEC2RescueForWindowsTool`.
4. Pilih dokumen `AWSSupport-RunEC2RescueForWindowsTool`, pilih Konten, lalu tampilkan JSON.

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 dengan AWS CLI, lihat [Referensi AWS CLI Perintah](#).

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" --comment "EC2Rescue offline volume xvdf" --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" --comment "EC2Rescue online log collection to S3" --parameters "Command=CollectLogs, Parameters='YOURS3BUCKETNAME'" --output text
```

Kumpulkan log dari volume instans Windows Amazon EC2 offline

Kumpulkan semua log dari volume offline yang dilampirkan ke instans Windows Amazon EC2 dan unggah ke Amazon S3 dengan URL yang telah ditentukan:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue offline log collection to S3" --parameters "Command=CollectLogs, Parameters=\"-Offline -BlockDeviceName xvdf - S3PreSignedUrl 'YOURS3PRESIGNEDURL'\\"" --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.

Atur ulang kata sandi Administrator lokal untuk instans online menggunakan AWS KMS key alias/ aws/ssm default:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --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" --comment "EC2Rescue online password reset" --parameters "Command=ResetAccess, Parameters=a133dc3c-a2g4-4fc6-a873-6c0720104bf0" --output text
```

Note

Dalam contoh ini, kunci KMS adalah a133dc3c-a2g4-4fc6-a873-6c0720104bf0.

Konsol Serial EC2 untuk instans Amazon EC2

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 [Konfigurasi akses ke Konsol Serial EC2](#).

Konsol serial dapat diakses menggunakan konsol EC2 atau AWS CLI.

Konsol serial tersedia tanpa biaya tambahan.

Topik

- [Prasyarat](#)
- [Konfigurasi akses ke Konsol Serial EC2](#)
- [Hubungkan ke Konsol Serial EC2](#)
- [Memutuskan koneksi dari Konsol Serial EC2](#)
- [Memecahkan masalah instans Amazon EC2 Anda menggunakan Konsol Serial EC2](#)

Prasyarat

Untuk terhubung ke Konsol Serial EC2 dan menggunakan alat yang Anda pilih untuk memecahkan masalah, prasyarat berikut harus tersedia:

- [Wilayah AWS](#)
- [Wavelength Zone dan Outposts AWS](#)
- [Zona Lokal](#)
- [Tipe instans](#)
- [Berikan akses](#)
- [Dukungan untuk klien berbasis peramban](#)
- [Status instans](#)
- [Amazon EC2 Systems Manager](#)
- [server sshd](#)
- [Konfigurasi alat pemecahan masalah yang Anda pilih](#)

Wilayah AWS

Didukung di semua Wilayah AWS kecuali Kanada Barat (Calgary).

Wavelength Zone dan Outposts AWS

Tidak didukung.

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
- Komputasi yang dipercepat: g5g.metal
- Memori yang dioptimalkan: u-6tb1.metal, u-9tb1.metal, u-12tb1.metal, u-18tb1.metal, u-24tb1.metal
- Windows

Semua instans virtual dibangun pada Nitro System. Instans bare metal tidak didukung.

Berikan akses

Anda harus menyelesaikan tugas konfigurasi untuk memberikan akses ke Konsol Serial EC2. Untuk informasi selengkapnya, lihat [Konfigurasi 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](#).

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 [Siklus hidup instans](#).

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 .

server sshd

Anda tidak perlu menginstal atau menjalankan server `sshd` pada instans.

Konfigurasi alat pemecahan masalah yang Anda pilih

Instans Linux

Untuk memecahkan masalah instance Linux Anda melalui konsol serial, Anda dapat menggunakan GRUB atau SysRq. Sebelum dapat menggunakan alat ini, Anda harus terlebih dahulu melakukan langkah-langkah konfigurasi pada setiap instans di tempat Anda akan menggunakannya.

Alat

- [Konfigurasi GRUB](#)
- [Konfigurasi SysRq](#)

Konfigurasi GRUB

Sebelum dapat menggunakan GRUB melalui konsol serial, Anda harus mengonfigurasi instans untuk menggunakan GRUB melalui konsol serial.

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](#)
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
```

CentOS

Untuk instans yang diluncurkan menggunakan CentOS AMI, GRUB dikonfigurasi 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"
```

Konfigurasi SysRq

Untuk mengkonfigurasi SysRq, Anda mengaktifkan SysRq perintah untuk siklus boot saat ini. Untuk membuat konfigurasi persisten, 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
```

Note

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 perintah login, masukkan nama pengguna dari pengguna berbasis kata sandi yang Anda [siapkan sebelumnya](#), lalu tekan Enter.
5. Pada perintah Password, masukkan kata sandi, lalu tekan Enter.

Instans Windows

Untuk memecahkan masalah instans Windows melalui konsol serial, Anda dapat menggunakan Konsol Admin Khusus (SAC). Sebelum dapat menggunakan SAC, Anda harus terlebih dahulu mengaktifkan SAC dan menu boot pada setiap instans di tempat Anda akan menggunakannya.

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 Windows di agen peluncuran Amazon EC2, lihat [the section called “Konfigurasi 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. [Connect](#) ke instans 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
```

Konfigurasi 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 instance Linux, Anda juga harus mengonfigurasi pengguna berbasis kata sandi pada setiap instance sehingga pengguna Anda dapat menggunakan konsol serial untuk pemecahan masalah.

Sebelum memulai, pastikan untuk memeriksa [prasyarat](#).

Topik

- [Tingkat akses ke Konsol Serial EC2](#)
- [Kelola akses akun ke Konsol Serial EC2](#)
- [Konfigurasi kebijakan IAM untuk akses Konsol Serial EC2](#)
- [Tetapkan 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 mereka. Untuk informasi selengkapnya, lihat [Konfigurasi 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 [Konfigurasi kebijakan IAM untuk akses Konsol Serial EC2](#).

Tingkat OS (hanya instance 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 [Tetapkan 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.

Topik

- [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.

```
{
  "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.

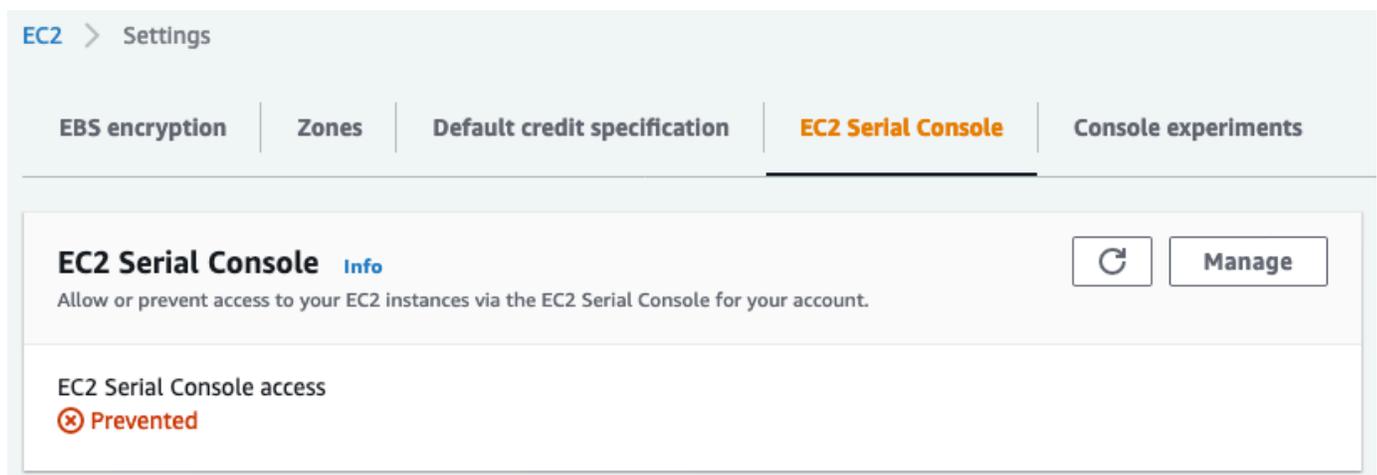
Melihat status akses akun ke konsol serial

Untuk melihat status akses akun ke konsol serial (konsol)

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Dasbor EC2.
3. Dari Atribut akun, pilih Konsol Serial EC2.

Bidang akses Konsol Serial EC2 menunjukkan apakah akses akun Diizinkan atau Dicegah.

Tangkapan layar berikut menunjukkan bahwa akun dicegah dari penggunaan konsol serial EC2.



Untuk melihat status akses akun ke konsol serial (AWS CLI)

Gunakan perintah [get-serial-console-access-status](#) untuk melihat status akses akun ke konsol serial.

```
aws ec2 get-serial-console-access-status --region us-east-1
```

Dalam output berikut, `true` menunjukkan bahwa akun diizinkan mengakses konsol serial.

```
{  
  "SerialConsoleAccessEnabled": true  
}
```

Memberikan akses akun ke konsol serial

Untuk memberikan akses akun ke konsol serial (konsol)

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Dasbor EC2.
3. Dari Atribut akun, pilih Konsol Serial EC2.
4. Pilih Kelola.
5. Untuk mengizinkan akses ke konsol serial EC2 dari semua instans di akun, pilih kotak centang Izinkan.
6. Pilih Perbarui.

Untuk memberikan akses akun ke konsol serial (AWS CLI)

Gunakan perintah [enable-serial-console-access](#) untuk mengizinkan akses akun ke konsol serial.

```
aws ec2 enable-serial-console-access --region us-east-1
```

Dalam output berikut, `true` menunjukkan bahwa akun diizinkan mengakses konsol serial.

```
{  
  "SerialConsoleAccessEnabled": true  
}
```

Menolak akses akun ke konsol serial

Untuk memberikan akses akun ke konsol serial (konsol)

1. Buka konsol Amazon EC2 di <https://console.aws.amazon.com/ec2/>.
2. Di panel navigasi kiri, pilih Dasbor EC2.
3. Dari Atribut akun, pilih Konsol Serial EC2.
4. Pilih Kelola.
5. Untuk mencegah akses ke konsol serial EC2 dari semua instans di akun, kosongkan kotak centang Izinkan.
6. Pilih Perbarui.

Untuk menolak akses akun ke konsol serial (AWS CLI)

Gunakan perintah [enable-serial-console-access](#) untuk mengizinkan akses akun ke konsol serial.

```
aws ec2 disable-serial-console-access --region us-east-1
```

Dalam output berikut, `false` menunjukkan bahwa akun ditolak untuk mengakses konsol serial.

```
{  
  "SerialConsoleAccessEnabled": false  
}
```

Konfigurasi 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 konfigurasi 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.

```
{
  "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:region:account-id: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.

```
{
  "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:region:account-id: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.

Kontrol akses berbasis atribut adalah strategi otorisasi yang mendefinisikan izin berdasarkan tag yang dapat dilampirkan ke pengguna dan sumber daya. AWS 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 penggunaan tag untuk mengontrol akses ke AWS sumber daya Anda, lihat [Mengontrol 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.

```
{
  "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:region:account-id:instance/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/SerialConsole":
            "${aws:PrincipalTag/SerialConsole}"
        }
      }
    }
  ]
}
```

Tetapkan kata sandi pengguna OS pada instance Linux

Note

Bagian ini hanya berlaku untuk 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

Petunjuk berikut hanya berlaku jika Anda meluncurkan instans menggunakan AMI Linux yang disediakan oleh AWS karena, 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 dikonfigurasi, 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 terhubung ke konsol serial dari instans EC2 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 Anda menggunakan Konsol Serial EC2](#).

Pertimbangan

- Hanya 1 koneksi konsol serial aktif yang didukung per instans.
- 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.
- Dibutuhkan waktu 30 detik untuk menghapus sesi setelah Anda memutuskan koneksi dari konsol serial untuk mengizinkan sesi baru.
- Port konsol serial yang didukung: `ttys0` (instance Linux) dan `COM1` (instance 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 [Konfigurasi akses ke Konsol Serial EC2](#).
- (Hanya instance Linux) Gunakan SysRq untuk terhubung ke konsol serial. SysRq tidak mengharuskan Anda terhubung melalui klien berbasis browser. Untuk informasi selengkapnya, lihat [Memecahkan masalah instance Linux Anda menggunakan SysRq](#).
- (Hanya instance 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 instance Anda dengan menggunakan SysRq (instans Linux), konsol EC2, atau AWS CLI Untuk informasi selengkapnya, lihat [Memecahkan masalah instance Linux Anda menggunakan SysRq](#) (contoh Linux) atau [Menyalakan ulang instans Anda](#).
5. (Hanya instance Linux) Pada **login** prompt, masukkan nama pengguna pengguna berbasis kata sandi yang Anda [atur sebelumnya](#), lalu tekan Enter.
 6. (Hanya contoh Linux) Pada Password prompt, masukkan kata sandi, lalu tekan Enter.

Anda sekarang masuk ke instans dan dapat menggunakan konsol serial untuk memecahkan masalah.

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.

Sebelum menghubungkan, pastikan Anda telah menyelesaikan [prasyarat](#).

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. Dibutuhkan waktu 30 detik untuk menghapus sesi setelah Anda memutuskan koneksi dari konsol serial untuk mengizinkan 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 instans dan port 0. Pada 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
```

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
```

Note

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 [Konfigurasi akses ke Konsol Serial EC2](#).
- (Hanya instance Linux) Gunakan SysRq untuk terhubung ke konsol serial. SysRq tidak mengharuskan Anda terhubung melalui SSH. Untuk informasi selengkapnya, lihat [Memecahkan masalah instance Linux Anda menggunakan SysRq](#).
- (Hanya instance 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 instance Anda dengan menggunakan SysRq (hanya instans Linux), konsol EC2, atau AWS CLI Untuk informasi selengkapnya, lihat

[Memecahkan masalah instance Linux Anda menggunakan SysRq](#) (hanya instance Linux) atau [Menyalakan ulang instans Anda](#).

5. (Hanya instance Linux) Pada **login** prompt, masukkan nama pengguna pengguna berbasis kata sandi yang Anda [atur sebelumnya](#), lalu tekan Enter.
6. (Hanya contoh Linux) Pada Password prompt, masukkan kata sandi, lalu tekan Enter.

Anda sekarang masuk ke instans dan dapat menggunakan konsol serial untuk memecahkan masalah.

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: EhWpk TzRtTY7trszz26xbb0/hvv9jrm7mczn0xW/d/0
US East (Northern Virginia)	us-east-1	serial-console.ec2-instance-connect.us-east-1.aws	SHA256: dxwn5mA/xadvmebzgeru5l2gx+yi5l Lucz0fmmW DiJa
US West (Northern California)	us-west-1	serial-console.ec2-instance-connect.us-west-1.aws	SHA256:OHldlcMET8u7QLSX3jmRTRAPFHVtqbyoLZBMUCqiH3Y
AS Barat (Oregon)	us-west-2	serial-console.ec2-instance-connect.us-west-2.aws	TqKasha256:emcie23bi6kuning dhavha1o2jxvuc HainqZcMwqNk
Afrika (Cape Town)	af-south-1	ec2-serial-console.af-south-1.api.aws	VePeSHA256:RMWWZ2f Juqzjo5jl

Nama Wilayah	Wilayah	Titik akhir	Sidik jari
			2k hlz21ed00biiwi lgXsczo
Asia Pasifik (Hong Kong)	ap-east-1	ec2-serial-console.ap-east-1.api.aws	SHA256XxCho: t0q1lpi z p7tkm2xxv ic9bj HplnAkjb FsjYnifk
Asia Pasifik (Hyderabad)	ap-south-2	ec2-serial-console.ap-south-2.api.aws	SHA256: wjgpbswv4 /shn+opit yj15dvw84 5jehdkrs ValoewAu
Asia Pasifik (Jakarta)	ap-southeast-3	ec2-serial-console.ap-southeast-3.api.aws	SHA 256:5 ZwgrCh +lfns32xitqI/4o0zi fbx4bzgsyfqy3o8mlk
Asia Pasifik (Melbourne)	ap-southeast-4	ec2-serial-console.ap-southeast-4.api.aws	sha256:avaq27h FgLvjn 5gtsshz0o v7h90p0gg46wfoet6z jvm
Asia Pasifik (Mumbai)	ap-south-1	serial-console.ec2-instance-connect.ap-south-1.aws	SHA256XcYmklq: OBL Hheblia H8iso51reztPism35b su40 RxEg
Asia Pacific (Osaka)	ap-northeast-3	ec2-serial-console.ap-northeast-3.api.aws	SHA256:am0/ jibk BnBu FnHr 9axsgev3g8tu/vvhfx e/3ucyjsq
Asia Pasifik (Seoul)	ap-northeast-2	serial-console.ec2-instance-connect.ap-northeast-2.aws	SHA256:FoqWXNX +DZ++GuNTztg9 PK49WYMqBX +FrcZM2dSrql

Nama Wilayah	Wilayah	Titik akhir	Sidik jari
Asia Pasifik (Singapura)	ap-southeast-1	serial-console.ec2-instance-connect.ap-southeast-1.aws	SHA256: plfnn7wncqdhx3qmwlu1gy/o8tux7l c6l45coy QgZua
Asia Pasifik (Sydney)	ap-southeast-2	serial-console.ec2-instance-connect.ap-southeast-2.aws	sha256:y FvMwuk9leuqjqtroxxzun+cw9/vse9w984cf5tgzo4
Asia Pasifik (Tokyo)	ap-northeast-1	serial-console.ec2-instance-connect.ap-northeast-1.aws	SHA256: rqfsdcztOfQawew trdv1t9em/hmrfqe+crlot5um4k
Kanada (Pusat)	ca-central-1	serial-console.ec2-instance-connect.ca-central-1.aws	ZwmpMwkps ha256:p2o2jo o6yw738fiothU 2gczYmMo7s4 TyEv
Tiongkok (Beijing)	cn-north-1	ec2-serial-console.cn-north-1.api.amazonwebservices.com.cn	SHA 256:2 GHVfY4h7U U3+wafuxD28V/gg LGGT+Y MeqjvSlgn gpg
Tiongkok (Ningxia)	cn-northwest-1	ec2-serial-console.cn-northwest-1.api.amazonwebservice.com.cn	OdVfsha256:tdgrnzkiqyebuho4szua09vwi5ryozgtogpwmim
Eropa (Frankfurt)	eu-central-1	serial-console.ec2-instance-connect.eu-central-1.aws	SHA256: ACMFS/YOl8amz1toe+bbnrjj3fy0k0de2c lcOd OIkXv
Eropa (Irlandia)	eu-west-1	serial-console.ec2-instance-connect.eu-west-1.aws	sha256:h2aagawo4hathhtm6ezs3bj7udguxi2q zawcw6e TrHj

Nama Wilayah	Wilayah	Titik akhir	Sidik jari
Eropa (London)	eu-west-2	serial-console.ec2-instance-connect.eu-west-2.aws	RnJgsha256:a69rd5ce/aeg4amm53i6lkd1z pvs/bcv3tppw2 8
Eropa (Milan)	eu-south-1	ec2-serial-console.eu-south-1.api.aws	sha256:lc0kov JnpgFy bvrxn0a7n99eclbxsx 95cuus7x7qk30
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: gocw2dfrlu669q NxqFx ecsr6fzuz/4f4n7t45 ZcwoEc
Eropa (Stockholm)	eu-north-1	serial-console.ec2-instance-connect.eu-north-1.aws	DvocDiSHA256:tkgffuvu gss3cu8gd l6w2ui32epnpkflwx 84
Eropa (Zürich)	eu-central-2	ec2-serial-console.eu-central-2.api.aws	SHA 256:8 ppx2mbmf6 0N WdCw M4/4oAxfu tqxwp6mk UlzKfw lfrz
Israel (Tel Aviv)	il-central-1	ec2-serial-console.il-central-1.api.aws	SHA256NvtYy: jr6q8v6knnpi8+qsfq 4dj5dimnmzptgwgsms U
Timur Tengah (Bahrain)	me-south-1	ec2-serial-console .me-south-1.api.aws	SHA256:npjllkhu2 QnLd uq2kvarso k5xvpjomrjkcbzcdqc 3k8

Nama Wilayah	Wilayah	Titik akhir	Sidik jari
Timur Tengah (UEA)	eu-central-1	ec2-serial-console .me-central-1.api.aws	FwPeyykSH A256:zpb5duKibz+L0 d b4mpbyhi/xzxnfsdk bvl e
Amerika Selatan (Sao Paulo)	sa-east-1	serial-console.ec2- instance-connect.sa- east-1.aws	SHA256:rd2+/32ognj ew1yvieme NaQz c +botbih62oqapdq1di
AWS GovCloud (AS-Timur)	us-gov-east-1	serial-console.ec2 -instance-connect. us-gov-east-1.amaz onaws.com	IkqnDcSHA256:tiwe1 9gwsoylclrtvu38yeeh +dh znmtebv f28
AWS GovCloud (AS-Barat)	us-gov-west-1	serial-console.ec2 -instance-connect. us-gov-west-1.amaz onaws.com	SHA256OIPf: kfofrwlaozfb+utbd3 brf8 8ngo2yylqx 5dq Zilw

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

- 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.
- Dibutuhkan waktu 30 detik untuk menghapus sesi setelah Anda memutuskan koneksi dari konsol serial untuk mengizinkan sesi baru.

Cara untuk memutuskan koneksi dari konsol serial bergantung pada klien.

Klien berbasis peramban

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 Anda 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.

Note

[Sebelum memulai, pastikan Anda telah menyelesaikan prasyarat.](#)

Instans Linux

Topik

- [Memecahkan masalah instans Linux Anda menggunakan GRUB](#)
- [Memecahkan masalah instance Linux Anda menggunakan SysRq](#)

Memecahkan masalah instans Linux Anda menggunakan GRUB

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 tersebut tidak dapat diakses melalui SSH normal, tetapi Anda dapat mengaksesnya melalui Konsol Serial EC2.

Single user mode

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.

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 perintah `login`, masukkan nama pengguna dari pengguna berbasis kata sandi yang Anda [siapkan sebelumnya](#), lalu tekan `Enter`.
9. Pada perintah `Password`, masukkan kata sandi, lalu tekan `Enter`.

Emergency mode

Mode darurat mirip dengan mode pengguna tunggal kecuali kernel berjalan pada runlevel terendah.

Untuk boot ke mode darurat, ikuti langkah yang sama seperti mode pengguna tunggal, tetapi pada langkah 6, tambahkan kata `emergency` sebagai ganti `single`.

Memecahkan masalah instance Linux Anda menggunakan SysRq

Kunci System Request (SysRq), yang kadang-kadang disebut sebagai SysRq “magic”, dapat digunakan untuk langsung mengirim perintah 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 selengkapnya, lihat [SysRq Kunci 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 menggunakannya 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, tekan `h`.

```
[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, tekan h.

```
[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 Windows 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 Windows di agen peluncuran Amazon EC2, lihat [the section called “Konfigurasi instance Windows”](#). Jika mengaktifkan SAC, Anda dapat menonaktifkannya nanti. Untuk informasi selengkapnya, lihat [Menonaktifkan SAC dan menu boot](#).

Topik

- [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>.

```

Computer is booting, SAC started and initialized.

Use the "ch -?" command for information about using channels.
Use the "?" command for general help.

SAC>?
EVENT: The CMD command is now available.
SAC_

```

- Untuk menampilkan perintah SAC, masukkan `?`, lalu tekan Enter.

Output yang diharapkan

```

SAC>?
ch          Channel management commands. Use ch -? for more help.
cmd        Create a Command Prompt channel.
d          Dump the current kernel log.
f          Toggle detailed or abbreviated tlist info.
? or help  Display this list.
i          List all IP network numbers and their IP addresses.
i <#> <ip> <subnet> <gateway> Set IPv4 addr., subnet and gateway.
id         Display the computer identification information.
k <pid>    Kill the given process.
l <pid>    Lower the priority of a process to the lowest possible.
lock      Lock access to Command Prompt channels.
m <pid> <MB-allow> Limit the memory usage of a process to <MB-allow>.
p         Toggle paging the display.
r <pid>    Raise the priority of a process by one.
s         Display the current time and date (24 hour clock used).
s mm/dd/yyyy hh:mm Set the current time and date (24 hour clock used).
t         Tlist.
restart   Restart the system immediately.
shutdown  Shutdown the system immediately.
crashdump Crash the system. You must have crash dump enabled.

```

- Untuk membuat saluran prompt perintah (seperti `cmd0001` atau `cmd0002`), masukkan `cmd`, lalu tekan Enter.
- Untuk melihat saluran prompt perintah, tekan ESC, lalu tekan TAB.

Output yang diharapkan

```
Name: Cmd0001
Description: Command
Type: VT-UTF8
Channel GUID: ef9f20a0-1287-11eb-82b0-0e4ba51872e5
Application Type GUID: 63d02271-8aa4-11d5-bccf-00b0d014a2d0

Press <esc><tab> for next channel.
Press <esc><tab>0 to return to the SAC channel.
Use any other key to view this channel.
```

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.

```
Please enter login credentials.
Username: Administrator
Domain : .
Password: *****
```

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.

```
Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>diskpart

Microsoft DiskPart version 10.0.17763.1

Copyright (C) Microsoft Corporation.
On computer: EC2AMAZ-ASR4SAI

DISKPART> list disk

   Disk ###  Status              Size               Free              Dyn  Gpt
   -----  -
   Disk 0    Online              30 GB               0 B
   Disk 1    Online              46 GB              46 GB

DISKPART>
```

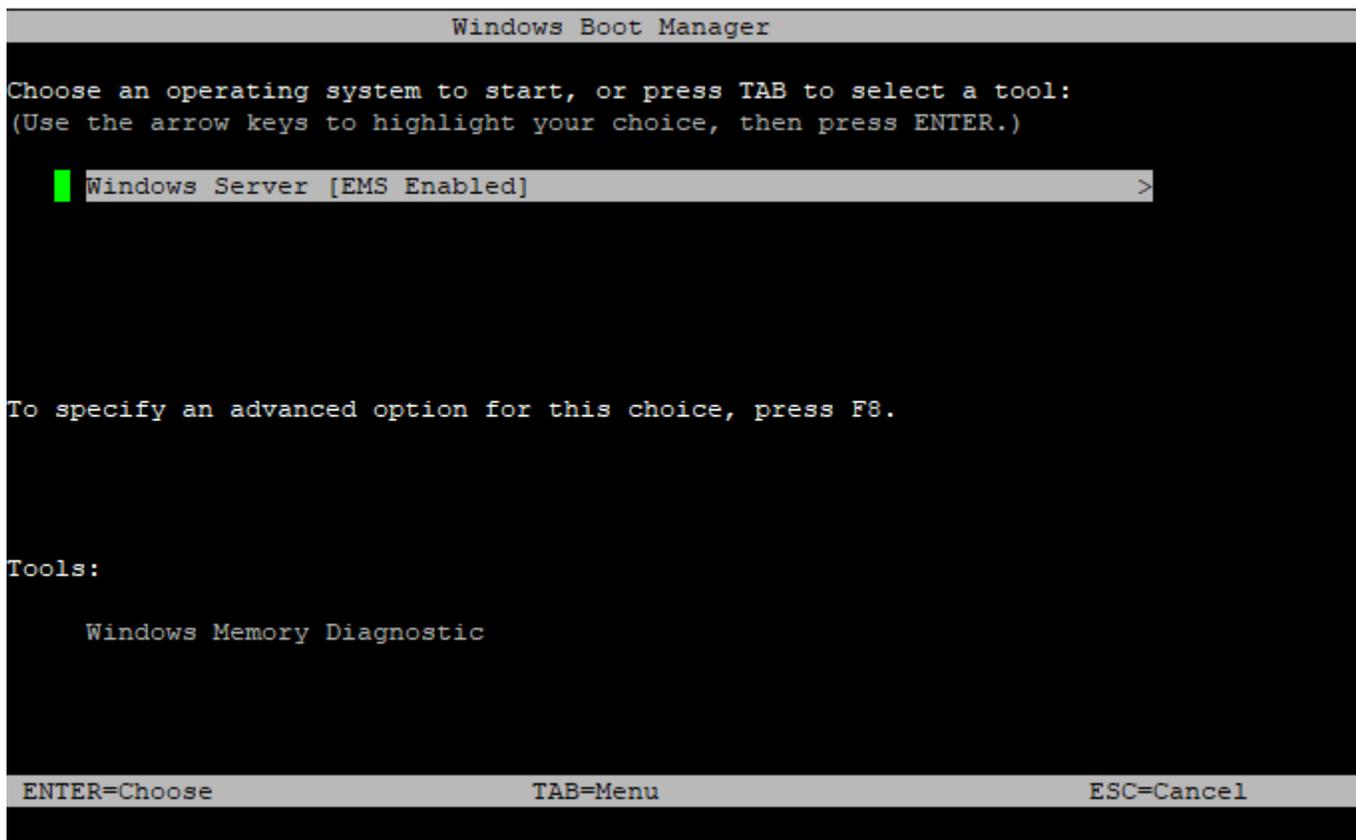
PowerShell juga dapat digunakan dari command prompt.

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 menu boot pada instans aktif dan dimulai ulang setelah terhubung melalui SSH, Anda akan melihat menu boot, seperti berikut.

A screenshot of the Windows Boot Manager interface. The title bar reads "Windows Boot Manager". The main text says "Choose an operating system to start, or press TAB to select a tool: (Use the arrow keys to highlight your choice, then press ENTER.)". A single option, "Windows Server [EMS Enabled]", is highlighted with a green bar on the left and a right-pointing arrow on the right. Below this, it says "To specify an advanced option for this choice, press F8." Under the heading "Tools:", the option "Windows Memory Diagnostic" is listed. At the bottom, a grey bar contains the instructions: "ENTER=Choose", "TAB=Menu", and "ESC=Cancel".

```
Windows Boot Manager

Choose an operating system to start, or press TAB to select a tool:
(Use the arrow keys to highlight your choice, then press ENTER.)

Windows Server [EMS Enabled] >

To specify an advanced option for this choice, press F8.

Tools:

Windows Memory Diagnostic

ENTER=Choose          TAB=Menu          ESC=Cancel
```

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.

Note

Tombol ESC tidak membawa Anda kembali ke menu utama karena Windows menunggu untuk melihat jika urutan keluar sedang berlangsung.

```
Advanced Boot Options

Choose Advanced Options for: Windows Server
(Use the arrow keys to highlight your choice.)

Repair Your Computer

Safe Mode
Safe Mode with Networking
Safe Mode with Command Prompt

Enable Boot Logging
Enable low-resolution video
Last Known Good Configuration (advanced)
Debugging Mode
Disable automatic restart on system failure
Disable Driver Signature Enforcement
Disable Early Launch Anti-Malware Driver

Start Windows Normally

Description: View a list of system recovery tools you can use to repair
startup problems, run diagnostics, or restore your system.

ENTER=Choose                                ESC=Cancel
```

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. [Connect](#) ke instans 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
```

Kirimkan interupsi diagnostik (untuk pengguna tingkat lanjut)

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 instance yang tidak dapat dijangkau atau tidak responsif untuk memicu kepanikan 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 instans berbasis Nitro, kecuali yang didukung oleh prosesor AWS Graviton. [Untuk informasi lebih lanjut, lihat contoh yang dibangun di atas Sistem AWS Nitro dan AWS Graviton.](#)

Prasyarat

Sebelum menggunakan interupsi diagnostik, Anda harus mengonfigurasi sistem operasi instans. Ini memastikan bahwa ia melakukan tindakan yang Anda butuhkan ketika kepanikan kernel (instance Linux) atau kesalahan berhenti (instance Windows) terjadi.

Instans Linux

Untuk mengonfigurasi Amazon Linux 2 agar membuat crash dump saat kepanikan kernel terjadi

1. Menghubungkan ke instans Anda.
2. Instal kexec dan kdump.

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

3. Konfigurasi 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 160MB, modifikasi file `grub` sebagai berikut:

```
GRUB_CMDLINE_LINUX_DEFAULT="crashkernel=160M 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=160M 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 Amazon Linux agar membuat crash dump saat kepanikan kernel terjadi

1. Terhubung ke instans Anda.
2. Instal `kexec` dan `kdump`.

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

3. Konfigurasi kernel untuk mencadangkan jumlah memori yang sesuai untuk kernel sekunder. Jumlah memori yang disimpan bergantung pada total memori yang tersedia pada instans Anda.

```
$ sudo grubby --args="crashkernel=memory_to_reserve" --update-kernel=ALL
```

Misalnya, untuk mencadangkan 160MB saat kernel mengalami crash, gunakan perintah berikut.

```
$ sudo grubby --args="crashkernel=160M" --update-kernel=ALL
```

4. 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
```

5. Boot ulang dan terhubung kembali ke instans Anda.
6. Verifikasi bahwa kernel telah dilakukan boot dengan parameter `crashkernel` yang benar.

```
$ grep crashkernel /proc/cmdline
```

Contoh output berikut mengindikasikan konfigurasi yang berhasil.

```
root=LABEL=/ console=tty1 console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295  
LANG=en_US.UTF-8 KEYTABLE=us crashkernel=160M
```

7. Verifikasi bahwa layanan kdump berjalan.

```
[ec2-user ~]$ sudo service kdump status
```

Jika layanan sedang berjalan, perintah tersebut akan mengembalikan respons `Kdump is operational`.

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 mengonfigurasi kernel agar macet ketika menerima NMI yang tidak dikenal dengan menyesuaikan file konfigurasi untuk sistem operasi Anda. Untuk informasi tentang cara mengonfigurasi 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, konfigurasi 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 AWS CLI atau Amazon EC2 API.

AWS CLI

Untuk mengirimkan interupsi diagnostik ke instans Anda (AWS CLI)

Gunakan perintah [send-diagnostic-interrupt](#) dan tentukan ID instans.

```
aws ec2 send-diagnostic-interrupt --instance-id i-1234567890abcdef0
```

PowerShell

Untuk mengirimkan interupsi diagnostik ke instans Anda (AWS Tools for Windows PowerShell)

Gunakan [Send-EC2DiagnosticInterruptcmdlet](#) dan tentukan ID instance.

```
PS C:\> Send-EC2DiagnosticInterrupt -InstanceId i-1234567890abcdef0
```

Riwayat dokumen

Tabel berikut menjelaskan penambahan penting pada Panduan Pengguna Amazon EC2 mulai tahun 2019. Kami juga sering memperbarui panduan untuk mengatasi umpan balik yang Anda kirimkan kepada kami.

Perubahan	Deskripsi	Tanggal
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.	Juni 5, 2024
Instans U7i-12tb, U7in-16tb, U7in-24tb, dan U7in-32TB	Jenis instans memori tinggi baru yang menampilkan prosesor Intel Xeon Scalable generasi ke-4.	28 Mei 2024
Kebijakan terkelola baru untuk Peluncuran Cepat EC2	Menambahkan EC2FastLaunchFullAccess kebijakan untuk melakukan tindakan API yang terkait dengan fitur Peluncuran Cepat EC2 dari sebuah instance.	14 Mei 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 tipe instance	Jam perangkat keras PTP sekarang tersedia pada jenis instans C7a, C7i, 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 berkemampuan VSS	Amazon EC2 VSS memiliki kebijakan terkelola IAM baru yang tersedia yang dapat Anda tambahkan ke peran profil instans untuk memastikan izin tetap up-to-date dan mengikuti praktik terbaik.	Maret 28, 2024
Jam perangkat keras PTP - AS Timur (Virginia N.)	Jam perangkat keras PTP sekarang tersedia di Wilayah AS Timur (Virginia N.).	Maret 26, 2024
Tetapkan IMDSv2 sebagai default akun	Anda dapat mengatur semua peluncuran instans EC2 baru di akun Anda untuk menggunakan Layanan Metadata Instans Versi 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.	7 Maret 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.	7 Maret 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 AWS dan lokal, lihat AWS Systems Manager Manajer Armada.](#)

Februari 12, 2024

[EC2 Instance Connect sudah diinstal sebelumnya di macOS AMI](#)

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 mempertimbangkan kasus penggunaan, tipe beban kerja, dan preferensi produsen CPU, serta cara Anda memprioritaskan 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

[Konsol-ke-Kode](#)

Konsol-ke-Kode dapat membantu untuk memulai dengan kode otomatisasi Anda. Konsol-ke-kode merekam tindakan konsol, lalu menggunakan AI generatif untuk menyarankan kode dalam format infrastruktur sebagai kode 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 an terbebannya 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 Anda, dan kemudian menggunakan informasi ini untuk mengoptimalkan pekerjaan HPC dan ML dengan menjalankannya pada instans yang secara fisik lebih dekat satu sama lain.

13 November 2023

EC2 Fast Launch berbagi dukungan AMI	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 akan 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	Sekarang Anda dapat menghibernasikan Instans Spot menggunakan pengalaman hibernasi dan keluarga instans yang sama yang saat ini tersedia untuk Instans Sesuai Permintaan.	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	Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans M7i dan M7i-flex.	22 Agustus 2023
EC2-Classic telah diusangkan	Dengan EC2-Classic, instans EC2 berjalan dalam jaringan datar tunggal 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 instans melalui SSH atau RDP tanpa mengharuskan instans memiliki alamat IPv4 publik.	13 Juni 2023

IMDS Package Analyzer	Anda sekarang dapat menggunakan IMDS Packet Analyzer untuk mengidentifikasi 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

Pembaruan sistem operasi macOS in place	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
Keamanan berbasis Virtualisasi Windows - Credential Guard	Anda dapat mengaktifkan Credential Guard, fitur Keamanan berbasis virtualisasi (VBS), pada instans Amazon EC2 yang didukung.	31 Januari 2023
Alias AMI dalam templat peluncuran	Anda dapat menentukan AWS Systems Manager parameter alih-alih ID AMI di templat peluncuran Anda untuk menghindari keharusan memperbarui templat 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 performa beban kerja basis data relasional intensif I/O Anda dan kurangi latensi tanpa berdampak negatif terhadap ketahanan data dengan pencegahan tumpang tindih, sebuah 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
Kunci aturan retensi Keranjang Sampah	Anda dapat mengunci aturan retensi untuk membantu melindunginya dari modifikasi dan penghapusan yang tidak disengaja atau berbahaya.	23 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 Akun AWS dan Anda tidak ingin lagi 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 Elastis 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	Konfigurasi 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 di AWS Outposts	Menambahkan strategi penyebaran host untuk grup penempatan di Outpost.	30 Juni 2022

Kunci syarat untuk Keranjang Sampah	Anda dapat menggunakan kunci syarat <code>rbn:Request/ResourceType</code> dan <code>rbn:Attribute/ResourceType</code> untuk memfilter akses pada permintaan Keranjang Sampah.	14 Juni 2022
Volume io2 Block Express	Anda dapat memodifikasi ukuran dan IOPS yang tersedia dari volume io2 Block Express dan Anda dapat mengaktifkannya untuk pemulihan snapshot cepat.	31 Mei 2022
Tuan Rumah 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	UEFI Secure Boot dibangun di atas proses boot aman lama Amazon EC2 dan menyediakan tambahan yang membantu pelanggan mengamankan perangkat lunak dari ancaman defense-in-depth yang bertahan selama reboot.	10 Mei 2022

NitroTPM	Nitro Trusted Platform Module (NitroTPM) adalah perangkat virtual yang disediakan oleh Sistem AWS Nitro 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 instans 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	<code>lastLaunchedTime</code> menunjukkan waktu AMI Anda terakhir kali digunakan untuk meluncurkan instans.	28 Februari 2022
Keranjang Sampah untuk AMI	Keranjang Sampah memungkinkan Anda memulihkan AMI yang terhapus secara tidak sengaja.	3 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 Kapasitas Sesuai Permintaan.	11 Januari 2022

Konfigurasi AMI Windows untuk peluncuran lebih cepat	Konfigurasi 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
Keranjang Sampah untuk snapshot Amazon EBS	Keranjang Sampah untuk snapshot Amazon EBS adalah fitur pemulihan snapshot yang memungkinkan Anda memulihkan snapshot yang terhapus secara tidak sengaja.	29 November 2021
Armada Spot launch-before-terminate	Armada Spot dapat mengakhiri Instans Spot yang menerima notifikasi penyeimbangan ulang setelah Instans Spot pengganti baru diluncurkan.	4 November 2021
Armada EC2 launch-before-terminate	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	Anda sekarang dapat berbagi AMI dengan AWS sumber daya berikut: organisasi dan unit organisasi (OU).	29 Oktober 2021
Skor penempatan Spot	Dapatkan rekomendasi untuk AWS Wilayah atau Availability Zone berdasarkan persyaratan kapasitas Spot Anda.	27 Oktober 2021
Pemilihan tipe instans berbasis atribut untuk Armada Spot	Tentukan atribut yang harus dimiliki oleh instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut.	27 Oktober 2021
Pemilihan tipe instans berbasis atribut untuk Armada EC2	Tentukan atribut yang harus dimiliki oleh instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut.	27 Oktober 2021
Armada Reservasi Kapasitas Sesuai Permintaan	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 Kapasitas Sesuai Permintaan yang ditargetkan	Armada EC2 dapat meluncurkan Instans Sesuai Permintaan ke Reservasi Kapasitas targeted.	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 AWS Wilayah dalam satu konsol.	1 September 2021
Dukungan pengusangan AMI untuk Amazon Data Lifecycle Manager	Kebijakan AMI yang didukung EBS Amazon Data Lifecycle Manager dapat mengusangkan AMI. Kebijakan AWSDataLifecycleManagerServiceRoleForAMIManagement AWS terkelola telah diperbarui untuk mendukung fitur ini.	23 Agustus 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 penggunaan berbasis Windows dan SQL Server per detik, dengan biaya minimum satu menit.	10 Juni 2021
Reservasi Kapasitas di 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 penggantian volume root untuk mengganti volume EBS guna menjalankan instans.	22 April 2021
Simpan dan pulihkan AMI menggunakan S3	Simpan AMI yang didukung EBS di S3 dan pulihkan dari S3 untuk memungkinkan penyalinan lintas-partisi AMI.	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 EC2 berbasis AMD dan Intel yang dipilih.	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 EventBridge untuk memantau peristiwa Spot Fleet	Buat EventBridge aturan yang memicu tindakan terprogram sebagai respons terhadap perubahan dan kesalahan status Armada Spot.	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 contoh: 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	Konfigurasi 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 an AWS Resource Groups untuk membuat koleksi logis dari Reservasi Kapasitas, dan kemudian menargetkan peluncuran instance ke grup tersebut.	29 Juli 2020
EC2Launch v2	Anda dapat menggunakan an 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 meluncurkan instance.	5 Mei 2020
Mengustomisasi notifikasi peristiwa terjadwal	Anda dapat mengustomisasi notifikasi peristiwa terjadwal untuk menyertakan tanda dalam notifikasi email.	4 Mei 2020

[Amazon Linux 2 Kernel Live Patching](#)

Kernel Live Patching untuk Amazon Linux 2 memungkinkan Anda menerapkan patch kerentanan keamanan dan bug kritis ke kernel Linux yang sedang berjalan, tanpa boot ulang atau gangguan pada aplikasi yang sedang berjalan.

28 April 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	Sekarang Anda 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 burstable di tingkat akun per AWS Wilayah.	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 Windows Sesuai Permintaan	Anda dapat menghibernasikan instans Windows Sesuai Permintaan.	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
Pembagian Reservasi Kapasitas Sesuai Permintaan	Sekarang Anda 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 instance 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.	8 Mei 2019
Peningkatan Otomatis Windows	Lakukan upgrade 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 [Riwayat dokumen di Panduan Jenis Instans Amazon EC2](#).

Sejarah untuk 2018 dan sebelumnya

Tabel berikut menjelaskan penambahan penting pada Panduan Pengguna Amazon EC2 di 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 guna menambahkan akselerasi yang didukung GPU untuk mengurangi biaya menjalankan inferensi deep learning.	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 permintaan 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 lanjut, sehingga memungkinkan Anda untuk	14 November 2018

Fitur	Versi API	Deskripsi	Tanggal rilis
		mengontrol jika dan waktu instans diluncurkan. Untuk informasi selengkapnya, lihat Tipe permintaan Armada 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 Mengoptimalkan opsi CPU .	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 Membuat templat peluncuran .	30 Oktober 2018
Reservasi Kapasitas Sesuai Permintaan	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 Reservasi Kapasitas Sesuai Permintaan .	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 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 Bring your own IP addresses (BYOIP) di 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 Alokasikan Host Khusus .	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 Menskalakan Armada Spot menggunakan penskalaan terjadwal .	20 September 2018
Strategi alokasi untuk Armada EC2	15-11-2015	Anda dapat menentukan apakah kapasitas Sesuai Permintaan dipenuhi berdasarkan harga (harga terendah dahulu) atau prioritas (prioritas tertinggi dahulu). Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Untuk informasi selengkapnya, lihat Strategi alokasi untuk Instans Spot .	26 Juli 2018

Fitur	Versi API	Deskripsi	Tanggal rilis
Strategi alokasi untuk Armada Spot	15-11-2015	Anda dapat menentukan apakah kapasitas Sesuai Permintaan dipenuhi berdasarkan an harga (harga terendah dahulu) atau prioritas (prioritas tertinggi dahulu). Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Untuk informasi selengkapnya, lihat Strategi alokasi untuk Instans Spot .	26 Juli 2018
Otomatisasikan 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 Membuat templat peluncuran .	11 Juli 2018
Menandai Host Khusus	15-11-2015	Anda dapat menandai Host Khusus. Untuk informasi selengkapnya, lihat Tandai Host Khusus .	3 Juli 2018
Mendapatkan output konsol terbaru	15-11-2015	Anda dapat mengambil output konsol terbaru untuk beberapa jenis instans saat Anda menggunakan perintah AWS CLI get-console-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 Mengoptimalkan opsi CPU .	8 Mei 2018
Armada EC2	15-11-2015	Anda dapat menggunakan Armada EC2 untuk meluncurkan grup instans di berbagai tipe instans EC2 dan Zona Ketersediaan, dan di seluruh model pembelian Instans Sesuai Permintaan, Instans Terpesan, dan Instans Spot. Untuk informasi selengkapnya, lihat Armada EC2 .	2 Mei 2018
Instans Sesuai Permintaan dalam Armada Spot	15-11-2015	Anda dapat menyertakan permintaan untuk kapasitas Sesuai Permintaan dalam permintaan Armada Spot untuk memastikan bahwa Anda tetap memiliki kapasitas instans. Untuk informasi selengkapnya, lihat 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 grup penempatan untuk instans .	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. Untuk informasi selengkapnya, lihat ID sumber daya .	9 Februari 2018

Fitur	Versi API	Deskripsi	Tanggal rilis
Peningkatan performa jaringan	15-11-2015	Instans di luar grup penempatan kluster 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. Untuk informasi selengkapnya, lihat 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 Atur waktu untuk instans Amazon EC2 .	29 November 2017
T2 Unlimited	15-11-2015	Instans T2 Unlimited dapat melonjak melampaui dasar selama yang dibutuhkan. Untuk informasi selengkapnya, lihat Instans 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 Meluncurkan sebuah instans dari templat peluncuran .	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. Untuk informasi selengkapnya, lihat Menghibernasi Instans Spot yang diinterupsi .	28 November 2017
Pelacakan target Armada Spot	15-11-2015	Anda dapat mengatur kebijakan penskalaan pelacakan target untuk Armada Spot. Untuk informasi selengkapnya, lihat Menskalakan Armada Spot menggunakan kebijakan pelacakan target .	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 penghunian VPC .	16 Oktober 2017
Penagihan per detik	15-11-2015	Amazon EC2 mengenakan biaya untuk penggunaan berbasis Linux 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. Untuk informasi selengkapnya, lihat Aturan-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. Untuk informasi selengkapnya, lihat Memulihkan alamat IP Elastis .	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

Fitur	Versi API	Deskripsi	Tanggal rilis
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 Memberikan izin untuk memberi tanda pada sumber daya saat sumber daya tersebut dibuat .	28 Maret 2017
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 Menentukan penghunian untuk Instans Spot Anda .	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

Fitur	Versi API	Deskripsi	Tanggal rilis
Penskalaan otomatis untuk Armada Spot		Anda sekarang dapat mengatur kebijakan penskalaan untuk Armada Spot. Untuk informasi selengkapnya, lihat Penskalaan otomatis untuk Armada Spot .	1 September 2016
Adaptor Jaringan Elastis (ENA)	01-04-2016	Anda sekarang dapat menggunakan ENA untuk jaringan yang ditingkatkan. Untuk informasi selengkapnya, lihat Dukungan jaringan yang ditingkatkan .	28 Juni 2016
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. Untuk informasi selengkapnya, lihat ID sumber daya .	23 Juni 2016
Salin snapshot Amazon EBS terenkripsi antar akun AWS	01-04-2016	Anda sekarang dapat menyalin snapshot EBS terenkripsi antar akun. AWS	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

Fitur	Versi API	Deskripsi	Tanggal rilis
Ditambahkan baru NetworkPacketsIn dan NetworkPacketsOut metrik untuk Amazon EC2		Ditambahkan baru NetworkPacketsIn dan NetworkPacketsOut metrik 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 CloudWatch metrik untuk Spot Fleet .	21 Maret 2016
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	Kami secara bertahap memperkenalkan ID yang lebih panjang untuk beberapa tipe sumber daya Amazon EC2 dan Amazon EBS. Selama periode keikutsertaan, Anda dapat mengaktifkan format ID yang lebih panjang untuk tipe sumber daya yang didukung. Untuk informasi selengkapnya, lihat ID sumber daya .	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 EC2-Classik tertaut dan instance dalam VPC menyelesaikan ke alamat IP pribadi dan bukan alamat IP publik.	11 Januari 2016

Fitur	Versi API	Deskripsi	Tanggal rilis
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 .	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
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 Strategi alokasi untuk Instans Spot .	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 Pembobotan instans Armada Spot .	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

Fitur	Versi API	Deskripsi	Tanggal rilis
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 Spot .	18 Mei 2015
Memigrasikan alamat IP Elastis ke EC2-Classic	15-04-2015	Anda dapat memigrasikan alamat IP Elastis yang telah dialokasikan untuk digunakan di EC2-Classic agar digunakan di VPC.	15 Mei 2015
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 Mengimpor VM sebagai Citra Menggunakan VM Import/Eksport di Panduan Pengguna VM Import/Export.	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

Fitur	Versi API	Deskripsi	Tanggal rilis
Pemulihan otomatis untuk instans EC2		<p>Anda dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2 dan memulihkan instans secara otomatis jika menjadi rusak karena kegagalan perangkat keras yang mendasarinya atau masalah yang memerlukan AWS keterlibatan untuk memperbaiki. Instans yang dipulihkan identik dengan instans asli, termasuk ID instans, alamat IP, dan semua metadata instans.</p> <p>Untuk informasi selengkapnya, lihat Ketahanan contoh.</p>	12 Januari 2015
ClassicLink	01-10-2014	ClassicLink memungkinkan Anda untuk menautkan instans EC2-Classic Anda ke VPC di akun Anda. Anda dapat mengaitkan grup keamanan VPC dengan instans EC2-Classic, yang memungkinkan komunikasi antara instans EC2-Classic dan instans di VPC Anda menggunakan alamat IP privat.	7 Januari 2015
Pemberitahuan pengakhiran Instans Spot.		<p>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.</p> <p>Untuk informasi selengkapnya, lihat Pemberitahuan interupsi Instans Spot.</p>	5 Januari 2015

Fitur	Versi API	Deskripsi	Tanggal rilis
Systems Manager untuk Microsoft SCVMM		Systems Manager untuk Microsoft SCVMM menyediakan easy-to-use antarmuka yang sederhana 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 di Referensi API Amazon EC2.	23 Oktober 2014
Ditambahkan 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 Log di AWS CLI, atau Logs CloudWatch 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

Fitur	Versi API	Deskripsi	Tanggal rilis
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 satu digit milidetik, kemampuan melonjak hingga 3.000 IOPS untuk waktu yang lama, dan performa dasar 3 IOPS/GiB. Ukuran volume SSD Tujuan Umum dapat bervariasi mulai 1 GiB hingga 1 TiB.	16 Juni 2014
AWS Paket Manajemen		AWS Management Pack sekarang mendukung untuk System Center Operations Manager 2012 R2.	22 Mei 2014
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 Pengguna VM Import/Export .	16 Desember 2013

Fitur	Versi API	Deskripsi	Tanggal rilis
Izin tingkat sumber daya 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 sebuah instance dari AWS Marketplace		Anda sekarang dapat meluncurkan instance dari AWS Marketplace menggunakan wizard peluncuran Amazon EC2. Untuk informasi selengkapnya, lihat Luncurkan sebuah AWS Marketplace instance .	11 November 2013
Wizard peluncuran baru		Terdapat wizard peluncuran EC2 yang baru dan didesain ulang. Untuk informasi selengkapnya, lihat Meluncurkan sebuah instans menggunakan wizard peluncuran instans lama .	10 Oktober 2013
Memodifikasi tipe instans dari 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 Menetapkan alamat IPv4 publik selama peluncuran instans .	20 Agustus 2013

Fitur	Versi API	Deskripsi	Tanggal rilis
Memberikan izin tingkat sumber daya	15-06-2013	Amazon EC2 mendukung Amazon Resource Name (ARN) dan kunci syarat baru. Untuk informasi selengkapnya, lihat Kebijakan IAM 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
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 AWS Wilayah dengan cepat dan mudah. Untuk informasi selengkapnya, lihat Menyalin AMI .	11 Maret 2013

Fitur	Versi API	Deskripsi	Tanggal rilis
Meluncurkan instans ke VPC default	01-02-2013	AWS Akun Anda mampu meluncurkan instans ke EC2-Classik atau VPC, atau hanya ke VPC, secara dasar. region-by-region Jika Anda hanya dapat meluncurkan instans ke dalam VPC, kami membuat VPC default untuk Anda. Saat meluncurkan instans, kami akan meluncurkannya ke dalam VPC default, kecuali jika Anda membuat VPC non-default dan menentukannya saat meluncurkan instans tersebut.	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
Status permintaan Instans Spot	01-10-2012	Status permintaan Instans Spot memudahkan untuk menentukan status permintaan Spot Anda.	14 Oktober 2012

Fitur	Versi API	Deskripsi	Tanggal rilis
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 dengan IOPS yang tersedia memberikan performa tinggi yang dapat diprediksi untuk beban kerja intensif I/O, seperti aplikasi basis data, 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: <ul style="list-style-type: none"> • AWS kunci akses untuk aplikasi yang berjalan di instans Amazon EC2. • Rotasi otomatis tombol AWS akses pada instans Amazon EC2. • Izin terperinci untuk aplikasi yang berjalan di instans Amazon EC2 yang membuat permintaan ke layanan Anda. AWS 	11 Juni 2012

Fitur	Versi API	Deskripsi	Tanggal rilis
Fitur Instans Spot yang memudahkan untuk memulai dan menangani potensi interupsi.		<p>Anda sekarang dapat mengelola Instans Spot sebagai berikut:</p> <ul style="list-style-type: none">• 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 selengkapnya, lihat Meluncurkan Instans Spot di Grup Auto Scaling dalam Panduan Pengguna Amazon EC2 Auto Scaling.• Dapatkan notifikasi saat instans diluncurkan atau diakhiri.• Gunakan AWS CloudFormation template 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	<p>Menambahkan dukungan untuk mengekspor instans Windows Server yang awalnya Anda impor ke EC2.</p> <p>Menambahkan dukungan untuk stempel waktu pada status instans dan status sistem untuk menunjukkan tanggal dan waktu ketika pemeriksaan status mengalami kegagalan.</p>	25 Mei 2012

Fitur	Versi API	Deskripsi	Tanggal rilis
Mengekspor instans EC2 dan stempel waktu dalam pemeriksaan status instans serta sistem untuk Amazon VPC	01-05-2012	Menambahkan dukungan untuk ekspor instans EC2 ke Citrix Xen, Microsoft Hyper-V, dan VMware vSphere. Menambahkan dukungan untuk stempel waktu dalam instans dan pemeriksaan status sistem.	25 Mei 2012
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

Fitur	Versi API	Deskripsi	Tanggal rilis
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 Anda .	16 November 2011
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 <code>ImportInstance</code> dan <code>ImportVolume</code> , yang sekarang akan melakukan pengunggahan gambar ke Amazon EC2 setelah membuat tugas impor. Selain itu, dengan diperkenalkannya <code>ResumeImport</code> , pengguna dapat memulai kembali unggahan yang belum selesai pada titik saat tugas tersebut terhenti.	15 September 2011

Fitur	Versi API	Deskripsi	Tanggal rilis
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 virtualisasi Citrix Xen dan Microsoft Hyper-V. Dengan rilis ini, VM Import sekarang mendukung format citra RAW, VHD, dan VMDK (kompatibel dengan VMware ESX). Untuk informasi selengkapnya, lihat Panduan Pengguna VM Import/Export .	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
Perubahan harga Zona Ketersediaan Instans Spot	15-05-2011	Menambahkan informasi tentang fitur harga Zona Ketersediaan Instans Spot. Dalam rilis ini, kami telah menambahkan opsi harga Zona Ketersediaan baru sebagai bagian dari informasi yang ditampilkan saat Anda membuat kueri untuk 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

Fitur	Versi API	Deskripsi	Tanggal rilis
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 hanya membayar untuk apa yang Anda gunakan, sekaligus mengisolasi instans komputasi Amazon EC2 Anda di tingkat perangkat keras. Untuk informasi selengkapnya, lihat Instans Khusus .	27 Maret 2011
Instans Cadangan diperbarui ke Konsol AWS Manajemen		Pembaruan pada Konsol AWS Manajemen 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 Cara menggunakan metadata instans dan Kategori metadata instans .	11 Maret 2011

Fitur	Versi API	Deskripsi	Tanggal rilis
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 AWS Management Console untuk memaksa pelepasan volume Amazon EBS dari sebuah instans.	23 Februari 2011
Perlindungan pengakhiran instans		Sekarang Anda dapat menggunakan AWS Management Console untuk mencegah instance dihentikan. Untuk informasi selengkapnya, lihat Aktifkan perlindungan pengakhiran .	23 Februari 2011
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 Pengguna VM Import/Export .	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 Membuat daftar dan memfilter sumber daya Anda dan Tandai sumber daya Amazon EC2 Anda .	19 September 2010

Fitur	Versi API	Deskripsi	Tanggal rilis
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
CloudWatch Pemantauan Amazon untuk Volume Amazon EBS		CloudWatch Pemantauan Amazon sekarang tersedia secara otomatis untuk volume Amazon EBS.	14 Juni 2010
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.