aws

Panduan Pengguna

Amazon EBS



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

Amazon EBS: Panduan Pengguna

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

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

Table of Contents

Apa itu Amazon EBS? 1
Fitur-fitur Amazon EBS 1
Layanan terkait 2
Mengakses Amazon EBS 3
Harga 4
Siapkan untuk Amazon EBS 5
Mendaftar untuk Akun AWS 5
(Opsional) Membuat dan menggunakan kunci yang dikelola pelanggan untuk enkripsi Amazon EBS 5
(Opsional) Aktifkan akses publik blokir untuk snapshot Amazon EBS 6
EBS volume 8
Fitur dan manfaat 9
Ketersediaan data 9
Persistensi data 10
Enkripsi data 11
Keamanan data 11
Snapshot 12
Fleksibilitas 12
Tipe volume EBS 13
Volume solid state drive (SSD) 13
Volume hard disk drive (HDD) 16
Volume generasi sebelumnya 17
Volume SSD Tujuan Umum 17
Volume SSD IOPS yang tersedia 23
Volume HDD dengan throughput yang dioptimalkan dan Cold HDD 27
Kendala volume EBS 37
Kapasitas penyimpanan 38
Pembatasan layanan 38
Skema partisi 39
Ukuran blok data 40
Volume EBS dan NVMe 43
Memetakan volume ke nama perangkat 44
I/O batas waktu operasi 48
Perintah Abort 49

Siklus hidup volume 49
Membuat volume 51
Bagikan volume 55
Salin volume 61
Lampirkan volume ke suatu instans 68
Melampirkan volume ke beberapa instans 71
Buat volume yang tersedia untuk digunakan 79
Lihat detail volume 94
Ubah volume 100
Melepaskan volume dari suatu instans 126
Menghapus volume 131
Ganti volume 132
Pemeriksaan status 135
Acara volume 138
Bekerja dengan volume yang terganggu 140
Auto-enable I/O 143
Pengujian kesalahan 145
Eksperimen yang tersedia 146
Pertimbangan-pertimbangan 146
Jeda injeksi I/O kesalahan 147
Injeksi latensi 148
Snapshot EBS 152
Cara kerja snapshot 153
Siklus hidup snapshot 157
Membuat snapshot 158
Melihat informasi snapshot 165
Menyalin snapshot 168
Membagikan snapshot 182
Mengarsipkan snapshot 190
Menghapus snapshot 225
Pemulihan snapshot cepat 230
Pertimbangan-pertimbangan 230
Harga dan Penagihan 231
Kredit pembuatan volume 231
Konfigurasikan pemulihan snapshot cepat 233
Periksa status pemulihan snapshot cepat 235

Lihat volume yang dipulihkan menggunakan pemulihan snapshot cepat 237
Kunci snapshot 238
Konsep 239
Pertimbangan-pertimbangan 242
Kontrol akses 243
Mengunci snapshot 247
Membuka kunci snapshot 250
Memperbarui pengaturan kunci snapshot 251
Monitor kunci snapshot 252
Memblokir akses publik untuk snapshot 255
Izin IAM 257
Konfigurasi akses publik blok 258
Lihat pengaturan akses publik blok 262
Nonaktifkan blokir akses publik 264
Memantau blok akses publik 267
Snapshot lokal pada Outposts 269
Pertanyaan umum 270
Prasyarat 272
Pertimbangan-pertimbangan 71
Mengendalikan akses dengan IAM 273
Bekerja dengan snapshot lokal 275
Snapshot lokal di Zona Lokal 281
Pertanyaan umum 270
Pertimbangan-pertimbangan 71
Mengendalikan akses dengan IAM 284
Enkripsi EBS 288
Cara kerja enkripsi EBS 288
Cara kerja enkripsi EBS saat snapshot dienkripsi 289
Cara kerja enkripsi EBS saat snapshot yang tidak terenkripsi 290
Bagaimana kunci KMS yang tidak dapat digunakan memengaruhi kunci data 290
Zona Lokal, Pos Terdepan, dan Zona Panjang Gelombang 291
Zona Lokal 291
Pos Terdepan dan Zona Panjang Gelombang 292
Persyaratan 292
Tipe volume yang mendukung 293
Tipe instans yang didukung 293

Izin untuk pengguna 293
Izin untuk instans 294
Aktifkan enkripsi secara default 295
Kunci enkripsi default 299
Enkripsi sumber daya EBS 300
Enkripsi volume kosong pada saat pembuatan 301
Mengenkripsi sumber daya yang tidak terenkripsi 301
Putar tombol KMS 302
Contoh 303
Mengembalikan volume yang tidak terenkripsi (enkripsi secara default tidak diaktifkan) 303
Mengembalikan volume yang tidak terenkripsi (enkripsi secara default diaktifkan) 304
Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan) 305
Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan) 305
Re-encrypt volume terenkripsi 306
Re-encrypt snapshot terenkripsi 306
Memigrasikan data antara volume terenkripsi dan tidak terenkripsi 307
Hasil enkripsi 308
Performa EBS 312
Kiat performa Amazon EBS 312
Gunakan EBS-optimized instance 312
Konfigurasikan bandwidth instance 313
Memahami cara menghitung performa 313
Memahami beban kerja Anda 313
Waspadai penalti performa saat menginisialisasi volume dari snapshot 313
Faktor yang dapat menurunkan performa HDD 314
Tingkatkan pembacaan ke depan untuk beban kerja throughput tinggi dan berat baca di st1 dan sc1 (Hanya instans Linux) 314
Gunakan kernel Linux modern (Hanya instans Linux) 315
Gunakan RAID 0 untuk memaksimalkan pemanfaatan sumber daya instans 316
Pantau kinerja volume Amazon EBS 316
Optimisasi EBS 316
Inisialisasi volume 317
Gunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume 318
Gunakan snapshot yang diaktifkan untuk pemulihan snapshot cepat 323
Inisialisasi volume secara manual 323
Memantau inisialisasi volume 327

Pembobotan bandwidth instans yang dapat dikonfigurasi 332
I/O karakteristik dan pemantauan 333
IOPS 334
Panjang antrean volume dan latensi 335
I/O batas throughput ukuran dan volume 336
Memantau I/O karakteristik menggunakan CloudWatch 337
Memantau statistik I/O kinerja waktu nyata 339
Sumber daya terkait 339
Konfigurasi RAID 339
Opsi konfigurasi RAID 340
Membuat array RAID 0 340
Buat snapshot volume dalam suatu array RAID 349
Tolok ukur volume EBS 350
Siapkan instans Anda 350
Pasang alat tolok ukur 352
Pilih panjang antrean volume 353
Nonaktifkan C-states 354
Lakukan benchmarking 355
Amazon Data Lifecycle Manager 359
Kuota 359
Cara kerjanya 360
Kebijakan 360
Jadwal kebijakan 362
Tanda sumber daya target 363
Snapshot 363
EBS-backed AMI 363
Tanda Amazon Data Lifecycle Manager 363
Kebijakan default vs kustom 364
Perbandingan kebijakan snapshot EBS 364
EBS-backed Perbandingan kebijakan AMI 367
Buat kebijakan default 369
Pertimbangan untuk kebijakan default 369
Buat kebijakan default untuk snapshot Amazon EBS 370
Buat kebijakan default untuk EBS-backed AMI 374
Aktifkan kebijakan default di seluruh akun dan Wilayah 378
Buat kebijakan kustom untuk snapshot 383

Membuat kebijakan siklus hidup snapshot 383
Pertimbangan untuk kebijakan siklus hidup snapshot 399
Sumber daya tambahan 406
Mengotomatiskan snapshot yang konsisten dengan aplikasi 406
Kasus penggunaan lain untuk skrip pra dan pasca 443
Cara kerja skrip pra dan pasca 451
Identifikasi snapshot yang dibuat dengan skrip pra dan pasca 455
Pantau skrip pra dan pasca 455
Buat kebijakan khusus untuk AMI 456
Membuat kebijakan siklus hidup AMI 456
Pertimbangan untuk kebijakan siklus hidup AMI 463
Sumber daya tambahan 467
Mengotomatiskan salinan snapshot lintas akun 467
Membuat kebijakan salinan snapshot lintas akun 468
Tentukan filter deskripsi snapshot 480
Pertimbangan untuk kebijakan penyalinan snapshot lintas akun 480
Sumber daya tambahan 481
Ubah kebijakan 481
Hapus kebijakan 484
Kontrol akses 485
AWS kebijakan yang dikelola 488
Peran layanan IAM 496
Memantau kebijakan 503
Konsol dan AWS CLI 503
Memantau kebijakan menggunakan EventBridge 503
Memantau kebijakan menggunakan CloudWatch 506
Pencatatan Panggilan API Menggunakan AWS CloudTrail 520
Titik akhir layanan 524
Titik akhir IPv4 525
Dual-stack Titik akhir (IPv4 dan IPv6) 525
Titik akhir FIPS 526
Menentukan titik akhir 526
Titik akhir VPC antarmuka 526
Pertimbangan untuk titik akhir Amazon EBS VPC 527
Membuat titik akhir VPC antarmuka untuk Amazon EBS 528
Pemecahan Masalah 528

Kesalahan: `Peran dengan nama sudah ada` ... 528
AWS Backup para Amazon EBS ... 530
API langsung Amazon EBS ... 531
Harga ... 532
Harga API ... 532
Biaya jaringan ... 532
Konsep ... 533
Snapshot ... 533
Blok ... 533
Indeks blok ... 533
Token blok ... 533
Checksum ... 534
Enkripsi ... 534
Tindakan API ... 534
Penandatanganan Tanda Tangan Versi 4 ... 535
Kontrol akses ... 535
Membaca snapshot ... 541
Mencantumkan blok dalam snapshot ... 542
Blok daftar yang berbeda antara dua snapshot ... 545
Dapatkan data blok dari snapshot ... 548
Menulis snapshot ... 549
Mulai snapshot ... 551
Menempatkan data ke dalam snapshot ... 553
Menyelesaikan snapshot ... 554
Hasil enkripsi ... 555
Hasil enkripsi: Snapshot induk yang tidak terenkripsi ... 556
Hasil enkripsi: Snapshot induk yang tidak terenkripsi ... 557
Hasil enkripsi: Tidak ada snapshot induk ... 558
Validasi data snapshot ... 559
Pastikan idempotensi ... 560
Kesalahan mencoba lagi ... 561
Optimalkan performa ... 564
Titik akhir layanan ... 565
Titik akhir IPv4 ... 566
Dual-stack Titik akhir (IPv4 dan IPv6) ... 566
Titik akhir FIPS ... 567

Menentukan titik akhir ... 568
Contoh kode SDK ... 569
StartSnapshot ... 570
PutSnapshotBlock ... 570
CompleteSnapshot ... 571
Titik akhir VPC antarmuka ... 572
Pertimbangan untuk titik akhir VPC API langsung EBS ... 573
Buat suatu titik akhir VPC antarmuka untuk API langsung EBS ... 574
CloudTrail log ... 574
Event data API langsung EBS di CloudTrail ... 576
Event manajemen API langsung EBS di CloudTrail ... 577
Contoh acara API langsung EBS ... 521
FAQ ... 583
Keranjang Sampah ... 586
Sumber daya yang didukung ... 587
Bagaimana cara kerjanya? ... 588
Pertimbangan-pertimbangan ... 589
Kuota ... 593
Layanan-layanan terkait ... 593
Harga ... 593
Kontrol akses ... 594
Izin untuk menggunakan Keranjang Sampah dan aturan retensi ... 595
Izin untuk menggunakan sumber daya di Keranjang Sampah ... 596
Kunci syarat untuk Keranjang Sampah ... 597
Buat aturan retensi ... 600
Perbarui aturan retensi ... 604
Aturan retensi kunci ... 606
Buka kunci aturan retensi ... 608
Menandai aturan retensi ... 609
Melihat tanda aturan retensi ... 610
Menghapus tanda dari aturan retensi ... 611
Hapus aturan retensi ... 612
Pulihkan snapshot yang dihapus ... 613
Izin untuk bekerja dengan snapshot di Keranjang Sampah ... 613
Lihat snapshot di Keranjang Sampah ... 615
Mengembalikan snapshot dari Keranjang Sampah ... 616

Pulihkan volume yang dihapus ... 617
Izin untuk bekerja dengan volume di Recycle Bin ... 617
Melihat volume di Recycle Bin ... 619
Kembalikan volume dari Recycle Bin ... 621
Pulihkan AMI yang dihapus ... 622
Izin untuk bekerja dengan AMI di Keranjang Sampah ... 622
Melihat AMI di Keranjang Sampah ... 623
Memulihkan AMI dari Keranjang Sampah ... 625
Memantau menggunakan EventBridge ... 626
RuleLocked ... 627
RuleChangeAttempted ... 627
RuleUnlockScheduled ... 628
RuleUnlockingNotice ... 629
RuleUnlocked ... 629
Memantau menggunakan CloudTrail ... 630
Informasi Recycle Bin di CloudTrail ... 520
Memahami entri file log Keranjang Sampah ... 632
Titik akhir layanan ... 645
Titik akhir IPv4 ... 566
Dual-stack Titik akhir (IPv4 dan IPv6) ... 646
Titik akhir FIPS ... 646
Menentukan titik akhir ... 647
Gunakan titik akhir antarmuka VPC ... 647
Buat titik akhir VPC antarmuka untuk Recycle Bin ... 648
Membuat kebijakan titik akhir VPC untuk Recycle Bin ... 648
Keamanan ... 650
Perlindungan data ... 650
Keamanan data Amazon EBS ... 652
Enkripsi saat istirahat dan dalam transit ... 652
Manajemen kunci KMS ... 652
Manajemen identitas dan akses ... 653
Audiens ... 653
Mengautentikasi dengan identitas ... 654
Mengelola akses menggunakan kebijakan ... 655
Bagaimana EBS bekerja dengan IAM ... 657
Contoh kebijakan IAM ... 663

Pemecahan Masalah 682
Validasi kepatuhan 684
Ketahanan data 685
Memantau 686
Amazon CloudWatch 687
Metrik untuk volume Amazon EBS 687
Metrik untuk snapshot Amazon EBS 711
Metrik untuk instans Nitro 711
Metrik untuk pemulihan snapshot cepat 716
Grafik konsol Amazon EC2 717
Amazon EventBridge 719
Peristiwa volume EBS 720
Acara inisialisasi volume EBS 726
Peristiwa modifikasi volume EBS 727
Peristiwa snapshot EBS 727
Peristiwa Arsip Snapshots EBS 735
Peristiwa pemulihan snapshot cepat EBS 736
Penggunaan AWS Lambda untuk menangani EventBridge peristiwa 737
Statistik kinerja terperinci EBS 740
Statistik 741
Mengakses statistik 743
Amazon GuardDuty 745
Laporan penagihan dan penggunaan 746
Snapshot 747
Penyimpanan volume 747
Kinerja yang disediakan 748
API langsung EBS 750
Pemulihan snapshot cepat (FSR) 750
Inventarisasi volume EBS Anda 750
volume 751
I/O kinerja 753
Isi volume data EBS Anda 754
Snapshot 754
Isi snapshot EBS Anda 755
Kuota 757
Riwayat dokumen 775

..... dcclxxxix

Apa itu Amazon Elastic Block Store?

Amazon Elastic Block Store (Amazon EBS) menyediakan sumber daya penyimpanan blok yang dapat diskalakan dan berkinerja tinggi yang dapat digunakan dengan instans Amazon Elastic Compute Cloud (Amazon EC2). Dengan Amazon Elastic Block Store, Anda dapat membuat dan mengelola sumber daya penyimpanan blok berikut:

- Volume Amazon EBS — Ini adalah volume penyimpanan yang Anda lampirkan ke instans Amazon EC2. Setelah Anda melampirkan volume ke instance, Anda dapat menggunakannya dengan cara yang sama seperti menggunakan hard drive lokal yang terpasang ke komputer, misalnya untuk menyimpan file atau menginstal aplikasi.
- Snapshot Amazon EBS — Ini adalah cadangan point-in-time volume Amazon EBS yang bertahan secara independen dari volume itu sendiri. Anda dapat membuat snapshot untuk membuat cadangan data pada volume Amazon EBS Anda. Anda kemudian dapat memulihkan volume baru dari snapshot tersebut kapan saja.

Topik

- Fitur-fitur Amazon EBS
- Layanan terkait
- Mengakses Amazon EBS
- Harga

Fitur-fitur Amazon EBS

Amazon EBS menyediakan fitur dan manfaat berikut:

- Beberapa jenis volume — Amazon EBS menyediakan beberapa jenis volume yang memungkinkan Anda mengoptimalkan kinerja penyimpanan dan biaya untuk berbagai aplikasi. Jenis volume dibagi menjadi dua kategori utama: SSD-backed penyimpanan untuk beban kerja transaksional, dan HDD-backed penyimpanan untuk beban kerja intensif throughput.
- Skalabilitas — Anda dapat membuat volume Amazon EBS dengan spesifikasi kapasitas dan kinerja yang memenuhi kebutuhan Anda. Seiring perubahan kebutuhan Anda, Anda dapat menggunakan operasi Volume Elastic untuk meningkatkan kapasitas atau menyesuaikan kinerja secara dinamis, tanpa downtime.

- Pencadangan dan pemulihan — Gunakan snapshot Amazon EBS untuk mencadangkan data yang disimpan di volume Anda. Anda kemudian dapat menggunakan snapshot tersebut untuk memulihkan volume secara instan atau memigrasikan data di seluruh AWS akun, Wil AWS ayah, atau Zona Ketersediaan.
- Perlindungan data — Gunakan enkripsi Amazon EBS untuk mengenkripsi volume Amazon EBS dan snapshot Amazon EBS Anda. Operasi enkripsi terjadi pada server yang meng-host instans Amazon EC2. Ini memastikan keamanan data-at-rest dan data-in-transit antara instance dan volume terlampirnya dan snapshot berikutnya.
- Ketersediaan dan daya tahan data — volume io2 Block Express memberikan daya tahan 99,999% dengan tingkat kegagalan tahunan 0,001%. Jenis volume lainnya memberikan daya tahan 99,8% hingga 99,9% dengan tingkat kegagalan tahunan 0,1% hingga 0,2%. Selain itu, data volume secara otomatis direplikasi di beberapa server di Zona Ketersediaan untuk mencegah hilangnya data dari kegagalan komponen tunggal mana pun.
- Pengarsipan data — EBS Snapshots Archive menyediakan tingkat penyimpanan berbiaya rendah untuk mengarsipkan salinan Snapshots EBS secara penuh dan tepat waktu. Gunakan Arsip jika Anda harus menyimpan snapshot selama 90 hari atau lebih untuk tujuan peraturan, kepatuhan, atau rilis proyek di masa mendatang.

Layanan terkait

Amazon EBS bekerja dengan layanan berikut:

- Amazon Elastic Compute Cloud — Layanan yang memungkinkan Anda meluncurkan dan mengelola mesin virtual (instans Amazon EC2) di AWS Cloud. Anda dapat melampirkan volume EBS ke instans tersebut dan menggunakannya dengan cara yang sama seperti Anda menggunakan hard drive lokal, misalnya untuk menyimpan file atau menginstal aplikasi. Untuk informasi selengkapnya, lihat Apa itu Amazon EC2?
- AWS Key Management Service— Layanan terkelola yang memungkinkan Anda membuat dan mengelola kunci kriptografi. Anda dapat menggunakan kunci AWS KMS kriptografi untuk mengenkripsi data yang disimpan di volume Amazon EBS Anda dan di snapshot Amazon EBS Anda. Untuk informasi selengkapnya, lihat Cara menggunakan AWS KMS Amazon EBS.
- Amazon Data Lifecycle Manager — Layanan terkelola yang mengotomatiskan pembuatan, penyimpanan, dan penghapusan snapshot EBS dan AMI. EBS-backed Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pencadangan untuk volume Amazon

EBS dan instans Amazon EC2 Anda. Untuk informasi selengkapnya, lihat Mengotomatiskan pencadangan dengan Amazon Data Lifecycle Manager.

- API langsung EBS — Layanan yang memungkinkan Anda membuat snapshot EBS, menulis data langsung ke snapshot Anda, membaca data dari snapshot Anda, dan mengidentifikasi perbedaan atau perubahan antara dua snapshot. Untuk informasi selengkapnya, lihat Menggunakan API langsung EBS untuk mengakses konten snapshot EBS.
- Recycle Bin — Layanan pemulihan data yang memungkinkan Anda memulihkan snapshot EBS dan EBS-backed AMI yang dihapus secara tidak sengaja. Untuk informasi selengkapnya, lihat Recycle Bin.

Mengakses Amazon EBS

Anda dapat membuat dan mengelola sumber daya Amazon EBS menggunakan antarmuka berikut:

Konsol Amazon EC2

Antarmuka web untuk membuat dan mengelola volume dan snapshot. Jika Anda telah mendaftar untuk AWS akun, Anda dapat mengakses konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

AWS Command Line Interface

Alat baris perintah yang memungkinkan Anda mengelola sumber daya Amazon EBS menggunakan perintah di shell baris perintah Anda. Hal ini didukung di Windows, Mac, dan Linux. Untuk informasi selengkapnya, lihat Panduan AWS Command Line Interface Pengguna dan perintah https://docs.aws.amazon.com/cli/latest/reference/ec2/ ec2.

Alat AWS untuk PowerShell

Satu set PowerShell modul yang memungkinkan Anda untuk membuat skrip operasi pada sumber daya Amazon EBS Anda dari baris PowerShell perintah. Untuk informasi selengkapnya, lihat Panduan Alat AWS untuk PowerShell Pengguna dan Referensi Alat AWS untuk PowerShell Cmdlet.

CloudFormation

AWS Layanan yang dikelola sepenuhnya yang memungkinkan Anda membuat template JSON atau YAML yang dapat digunakan kembali yang menggambarkan AWS sumber daya Anda, lalu menyediakan dan mengonfigurasi sumber daya tersebut untuk Anda. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS CloudFormation.

API Kueri Amazon EC2

Amazon EC2 Query API menyediakan permintaan HTTP atau HTTPS yang menggunakan kata kerja HTTP `GET` atau `POST` dan parameter kueri bernama`Action`. Untuk informasi selengkapnya, lihat Refer ensi API Amazon EC2.

AWS SDK

Language-specific API yang memungkinkan Anda membangun aplikasi yang terintegrasi dengan AWS layanan. AWS SDK tersedia untuk banyak bahasa pemrograman populer. Untuk informasi selengkapnya, lihat Alat untuk Membangun AWS.

Harga

Dengan Amazon EBS, Anda hanya membayar atas apa yang Anda berikan. Untuk informasi selengkapnya, lihat harga Amazon EBS.

Siapkan untuk Amazon EBS

Selesaikan tugas di bagian ini untuk menyiapkan untuk bekerja dengan sumber daya Amazon EBS.

Tugas

- Mendaftar untuk Akun AWS
- (Opsional) Membuat dan menggunakan kunci yang dikelola pelanggan untuk enkripsi Amazon EBS
- (Opsional) Aktifkan akses publik blokir untuk snapshot Amazon EBS

Mendaftar untuk Akun AWS

Untuk memulainya AWS, Anda membutuhkan Akun AWS. Untuk informasi tentang membuat Akun AWS, lihat Mem ulai dengan an Akun AWS di Panduan AWS Account Management Referensi.

(Opsional) Membuat dan menggunakan kunci yang dikelola pelanggan untuk enkripsi Amazon EBS

Enkripsi Amazon EBS adalah solusi enkripsi yang menggunakan kunci AWS KMS kriptografi untuk mengenkripsi volume Amazon EBS dan snapshot Amazon EBS Anda. Amazon EBS secara otomatis membuat kunci KMS ter AWS kelola unik untuk enkripsi Amazon EBS di setiap Wilayah. Kunci KMS ini memiliki alias `aws/ebs`. Anda tidak dapat memutar kunci KMS default atau mengelola izinnya. Untuk fleksibilitas dan kontrol yang lebih besar atas kunci KMS yang digunakan untuk enkripsi Amazon EBS, Anda dapat mempertimbangkan untuk membuat dan menggunakan kunci yang dikelola pelanggan.

Cara membuat dan menggunakan kunci yang dikelola pelanggan untuk enkripsi Amazon EBS

1. Buat kunci KMS enkripsi simetris.
2. Pilih kunci KMS sebagai kunci KMS default untuk enkripsi Amazon EBS.
3. Berikan pengguna izin untuk menggunakan kunci KMS untuk enkripsi Amazon EBS.

(Opsional) Aktifkan akses publik blokir untuk snapshot Amazon EBS

Untuk mencegah berbagi snapshot secara publik, Anda sekarang dapat mengaktifkan blokir akses publik untuk snapshot. Setelah Anda mengaktifkan blokir akses publik untuk snapshot di Wilayah, setiap upaya untuk membagikan snapshot secara publik di Wilayah tersebut akan diblokir secara otomatis. Pengaktifan ini dapat membantu Anda meningkatkan keamanan snapshot dan untuk melindungi data snapshot Anda dari akses yang tidak terotorisasi atau tidak diinginkan.

Untuk informasi selengkapnya, lihat Blokir akses publik untuk snapshot Amazon EBS.

Console

Untuk mengaktifkan akses publik blokir untuk snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2, lalu di Atribut akun (di sisi kanan), pilih Perlindungan dan keamanan data.
3. Di bagian Blokir akses publik untuk snapshot EBS, pilih Kelola.
4. Pilih Blokir akses publik dan kemudian pilih salah satu opsi berikut:
   - Blokir semua akses publik — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
   - Blokir semua pembagian publik — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk umum.
5. Pilih Perbarui.

AWS CLI

Untuk mengaktifkan akses publik blokir untuk snapshot

Gunakan perintah enable-snapshot-block-public-access. Untuk `--state`, tentukan salah satu dari nilai-nilai berikut:

- `block-all-sharing` — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
- `block-new-sharing` — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.

```
aws ec2 enable-snapshot-block-public-access --state block-new-sharing
```

PowerShell

Untuk mengaktifkan akses publik blokir untuk snapshot

Gunakan Enable-EC2SnapshotBlockPublicAccess cmdlet. Untuk `-State`, tentukan salah satu dari nilai-nilai berikut:

- `block-all-sharing` — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
- `block-new-sharing` — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.

```
Enable-EC2SnapshotBlockPublicAccess -State block-new-sharing
```

Volume Amazon EBS

Volume Amazon EBS adalah perangkat penyimpanan tingkat blok yang tahan lama yang dapat Anda pasang ke instans Anda. Setelah Anda memasang volume ke suatu instans, Anda dapat menggunakannya seperti Anda menggunakan hard drive fisik. Volume EBS bersifat fleksibel. Untuk volume generasi saat ini yang dipasang ke tipe instans generasi saat ini, Anda dapat secara dinamis meningkatkan ukuran, mengubah kapasitas IOPS yang tersedia, dan mengubah tipe volume pada volume produksi langsung.

Anda dapat menggunakan volume EBS sebagai penyimpanan utama untuk data yang memerlukan pembaruan rutin, seperti drive sistem untuk instans atau penyimpanan aplikasi basis data. Anda juga dapat menggunakannya untuk aplikasi yang membutuhkan banyak throughput dengan melakukan pemindaian disk secara terus-menerus. Volume EBS tetap bertahan secara independen dari siklus hidup instans EC2.

Anda dapat memasang beberapa volume EBS ke suatu instans. Volume dan instans harus berada dalam Zona Ketersediaan yang sama. Bergantung pada volume dan jenis instans, Anda dapat menggunakan Multi-Attach untuk memasang volume ke beberapa instans secara bersamaan.

Amazon EBS menyediakan tipe volume berikut ini: SSD Tujuan Umum (`gp2` dan `gp3`), SSD IOPS yang Tersedia (`io1` dan `io2`), HDD yang Dioptimalkan untuk Throughput (`st1`), dan Cold HDD (`sc1`), dan Magnetik (`standard`). Semuanya berbeda dalam karakteristik dan harga performa, memungkinkan Anda menyesuaikan performa dan biaya penyimpanan sesuai kebutuhan aplikasi Anda. Untuk informasi selengkapnya, lihat Tipe volume Amazon EBS.

Akun Anda memiliki batas total penyimpanan yang tersedia untuk Anda. Untuk informasi selengkapnya tentang pembatasan ini, dan cara meminta peningkatan dalam pembatasan Anda, lihat Titik akhir dan kuota Amazon EBS.

Volume EBS yang dikelola dikelola oleh penyedia layanan, seperti Mode Otomatis Amazon EKS. Anda tidak dapat secara langsung mengubah pengaturan volume EBS terkelola. Volume EBS yang dikelola diidentifikasi dengan nilai sebenarnya di bidang Terkelola. Untuk informasi selengkapnya, lihat Instans terkelola Amazon EC2.

Untuk informasi selengkapnya tentang harga, lihat Harga Amazon EBS.

Daftar Isi

- Fitur dan manfaat volume Amazon EBS

- Tipe volume Amazon EBS
- Batasan volume Amazon EBS
- Volume dan NVMe Amazon EBS
- Siklus hidup volume Amazon EBS
- Mengganti volume Amazon EBS menggunakan snapshot
- Pemeriksaan status volume Amazon EBS
- Pengujian kesalahan pada Amazon EBS

Fitur dan manfaat volume Amazon EBS

Volume EBS memberikan manfaat yang tidak disediakan oleh volume penyimpanan instans.

Manfaat

- Ketersediaan data
- Persistensi data
- Enkripsi data
- Keamanan data
- Snapshot
- Fleksibilitas

Ketersediaan data

Saat Anda membuat volume EBS, volume itu secara otomatis direplikasi dalam Zona Ketersediaannya untuk mencegah kehilangan data karena kegagalan komponen perangkat keras apa pun. Anda dapat memasang volume EBS ke instans EC2 apa pun di Zona Ketersediaan yang sama. Setelah Anda memasang volume, perangkat muncul sebagai perangkat blok asli yang serupa dengan hard drive atau perangkat fisik lainnya. Pada saat itu, instansnya dapat berinteraksi dengan volume sama seperti dengan drive lokal. Anda dapat terhubung ke instance dan memformat volume EBS dengan sistem file, seperti `Ext4` untuk instance Linux atau `NTFS` untuk instance Windows, dan kemudian menginstal aplikasi.

Jika Anda melampirkan beberapa volume ke perangkat yang telah Anda beri nama, Anda dapat membagi data di seluruh volume untuk meningkatkan kinerja I/O dan throughput.

Anda dapat melampirkan `io1` dan volume `io2` EBS hingga 16 Nitro-based instans. Untuk informasi selengkapnya, lihat Melampirkan volume EBS ke beberapa instans EC2 menggunakan Multi-Attach. Jika tidak, Anda dapat memasang volume EBS ke satu instans.

Anda bisa mendapatkan data pemantauan untuk volume EBS Anda, termasuk volume perangkat root untuk EBS-backed instans, tanpa biaya tambahan. Untuk informasi selengkapnya tentang cara memantau metrik, lihat CloudWatch Metrik Amazon untuk Amazon EBS. Untuk informasi tentang pelacakan status volume Anda, lihat Acara EventBridge Amazon untuk Amazon EBS.

Persistensi data

Volume EBS adalah penyimpanan off-instans yang dapat bertahan secara terpisah dari kehidupan sebuah instans. Anda terus membayar penggunaan volume selama data tetap ada.

Volume EBS yang dipasang ke instans yang berjalan dapat secara otomatis terlepas dari instans dengan datanya utuh saat instans dihentikan jika Anda menghapus tanda centang pada kotak centang Hapus saat Pengakhiran saat Anda mengonfigurasi volume EBS untuk instans Anda di konsol EC2. Volume kemudian dapat disambungkan kembali ke instans baru, memungkinkan pemulihan cepat. Jika kotak centang untuk Hapus saat Pengakhiran dicentang, volume akan dihapus pada saat pengakhiran instans EC2. Jika Anda menggunakan EBS-backed instance, Anda dapat menghentikan dan memulai ulang instance tersebut tanpa mempengaruhi data yang disimpan dalam volume terlampir. Volume tetap terpasang selama siklus stop-start. Hal ini memungkinkan Anda untuk memproses dan menyimpan data pada volume Anda secara tidak terbatas, hanya menggunakan sumber daya pemrosesan dan penyimpanan saat diperlukan. Data tetap ada pada volume hingga volume dihapus secara eksplisit. Penyimpanan blok fisik yang digunakan oleh volume EBS yang dihapus ditulis dengan nol atau data pseudorandom kriptografis sebelum dialokasikan ke volume baru. Jika Anda berurusan dengan data sensitif, Anda harus mempertimbangkan untuk mengenkripsi data Anda secara manual atau menyimpan data pada volume yang dilindungi oleh enkripsi Amazon EBS. Untuk informasi selengkapnya, lihat Enkripsi EBS Amazon.

Secara default, volume EBS utama yang dibuat dan dipasang pada suatu instans saat peluncuran akan dihapus bila instans tersebut diakhiri. Anda dapat mengubah perilaku ini dengan mengubah nilai bendera `DeleteOnTermination` ke `false` saat Anda meluncurkan instansnya. Nilai yang dimodifikasi ini menyebabkan volume berlanjut bahkan setelah instans dihentikan, dan memungkinkan Anda untuk memasang volume ke instans lain.

Secara default, volume EBS utama yang dibuat dan dipasang pada suatu instans saat peluncuran akan dihapus bila instans tersebut dihentikan. Anda dapat mengubah perilaku ini dengan mengubah

nilai bendera `DeleteOnTermination` ke `true` saat Anda meluncurkan instansnya. Nilai yang dimodifikasi ini menyebabkan volume dihapus ketika instans diakhiri.

Enkripsi data

Untuk enkripsi data yang disederhanakan, Anda dapat membuat volume EBS terenkripsi dengan fitur enkripsi Amazon EBS. Semua tipe volume EBS mendukung enkripsi. Anda dapat menggunakan volume EBS terenkripsi untuk memenuhi berbagai persyaratan enkripsi data-at-rest untuk regulated/audited data dan aplikasi. Enkripsi Amazon EBS menggunakan algoritma Standar Enkripsi Lanjutan 256-bit (AES-256) dan infrastruktur kunci. Amazon-managed Enkripsi terjadi pada server yang manjadi host instans EC2 dengan memberikan enkripsi data bergerak dari instans EC2 ke penyimpanan Amazon EBS. Untuk informasi selengkapnya, lihat Enkripsi EBS Amazon.

Enkripsi Amazon EBS digunakan AWS KMS keys saat membuat volume terenkripsi dan snapshot apa pun yang dibuat dari volume terenkripsi Anda. Saat pertama kali Anda membuat volume EBS terenkripsi di Wilayah, kunci KMS ter AWS kelola default dibuat untuk Anda secara otomatis. Kunci ini digunakan untuk enkripsi Amazon EBS kecuali Anda membuat dan menggunakan kunci yang dikelola pelanggan. Membuat kunci yang dikelola pelanggan Anda sendiri memberi Anda lebih banyak fleksibilitas, termasuk kemampuan untuk membuat, memutar, menonaktifkan, menentukan kontrol akses, dan mengaudit kunci enkripsi yang digunakan untuk melindungi data Anda. Untuk informasi selengkapnya, lihat Panduan Developer AWS Key Management Service.

Keamanan data

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.

Snapshot

Amazon EBS memberikan kemampuan untuk membuat snapshot (cadangan) dari volume EBS apa pun dan menulis salinan data dalam volume ke Amazon S3, yang menyimpan salinan itu secara redundan di beberapa Zona Ketersediaan. Volume tidak perlu dipasang ke instans yang sedang berjalan untuk mengambil snapshot. Saat Anda melanjutkan menulis data ke volume, Anda dapat membuat snapshot volume secara berkala untuk digunakan sebagai garis dasar untuk volume baru. Snapshot ini dapat digunakan untuk membuat banyak volume EBS baru atau memindahkan volume di seluruh Zona Ketersediaan. Snapshot volume EBS terenkripsi secara otomatis dienkripsi.

Saat Anda membuat volume baru dari snapshot, itu adalah salinan persis dari volume asli pada saat snapshot diambil. Volume EBS yang dibuat dari snapshot terenkripsi dienkripsi secara otomatis dienkripsi. Dengan menentukan Zona Ketersediaan yang berbeda, Anda dapat menggunakan fungsionalitas ini untuk membuat volume duplikat dalam zona tersebut. Snapshot dapat dibagikan dengan AWS akun tertentu atau dipublikasikan. Saat Anda membuat snapshot, Anda dikenai biaya di Amazon S3 berdasarkan ukuran data yang dicadangkan, bukan ukuran volume sumber. Snapshot berikutnya dengan volume yang sama adalah snapshot inkremental. Snapshot tersebut hanya menyertakan data yang diubah dan baru yang ditulis ke volume sejak snapshot terakhir dibuat, dan Anda hanya dikenai biaya untuk data yang diubah dan baru ini.

Snapshot adalah pencadangan bertahap, yang berarti hanya blok pada volume yang berubah setelah snapshot terbaru Anda disimpan. Jika Anda memiliki volume dengan 100 GiB data, tetapi hanya 5 GiB data telah berubah sejak snapshot terakhir Anda, hanya 5 GiB dari data yang dimodifikasi ditulis ke Amazon S3. Meskipun snapshot disimpan secara bertahap, proses penghapusan snapshot dirancang agar Anda hanya mempertahankan snapshot terbaru.

Untuk membantu mengategorikan dan mengelola volume dan snapshot, Anda dapat menandainya dengan metadata pilihan Anda.

Untuk mencadangkan volume Anda secara otomatis, Anda dapat menggunakan Amazon Data Lifecycle Manager atau AWS Backup.

Fleksibilitas

Volume EBS mendukung perubahan konfigurasi langsung saat berada di produksi. Anda dapat memodifikasi tipe volume, ukuran volume, dan kapasitas IOPS tanpa gangguan layanan. Untuk informasi selengkapnya, lihat Memodifikasi volume Amazon EBS menggunakan operasi Volume Elastic.

Tipe volume Amazon EBS

Amazon EBS menyediakan tipe volume berikut, yang berbeda dalam karakteristik performa dan harga, sehingga Anda dapat menyesuaikan performa dan biaya penyimpanan dengan kebutuhan aplikasi Anda.

> **Important**
>
> Ada beberapa faktor yang dapat mempengaruhi kinerja volume EBS, seperti konfigurasi instans, I/O karakteristik, dan permintaan beban kerja. Untuk sepenuhnya menggunakan IOPS yang disediakan pada volume EBS, gunakan instans yang dioptimalkan EBS. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-optimized.html Untuk informasi selengkapnya tentang memaksimalkan volume EBS Anda, lihat Kinerja volume Amazon EBS.

Untuk informasi selengkapnya tentang harga, lihat Harga Amazon EBS.

Tipe volume

- Volume solid state drive (SSD)
- Volume hard disk drive (HDD)
- Volume generasi sebelumnya

Volume solid state drive (SSD)

SSD-backed volume dioptimalkan untuk beban kerja transaksional yang melibatkan read/write operasi yang sering dengan I/O ukuran kecil, di mana atribut kinerja dominan adalah IOPS. SSD-backed Jenis volume termasuk SSD Tujuan Umum dan SSD IOPS Provisioned. Berikut ini adalah ringkasan kasus penggunaan dan karakteristik SSD-backed volume.

	Volume SSD Tujuan Umum Amazon EBS		Volume SSD IOPS yang disediakan Amazon EBS	
Tipe volume	`gp3`[6]	`gp2`	`io2`Blok Ekspres	`io1`
Daya tahan	Daya tahan 99,8% - 99,9% (tingkat kegagalan tahunan 0,1% - 0,2%)		Daya tahan 99,999% (tingkat	Daya tahan 99,8% - 99,9% (tingkat

	Volume SSD Tujuan Umum Amazon EBS		Volume SSD IOPS yang disediakan Amazon EBS	
			kegagalan tahunan 0,001%)	kegagalan tahunan 0,1% - 0,2%)
Kasus penggunaan	• Beban kerja transaksional • Desktop virtual • Medium-sized, database instance tunggal • Low-latency aplikasi interaktif • Volume boot • Lingkungan pengembangan dan pengujian		Beban kerja yang membutuhkan: • Latensi sub-milidetik yang konsisten dengan latensi rata-rata di bawah 500 mikrodetik 5 • Performa IOPS yang berkelanjutan • Lebih dari 80.000 IOPS atau 2.000 throughput MiB/s	• Beban kerja yang memerlukan performa IOPS berkelanjutan atau lebih dari 16.000 IOPS • I/O-intensive beban kerja database
Ukuran volume	1 GiB - 64 TiB	1 GiB - 16 TiB	4 GiB - 64 TiB	4 GiB - 16 TiB
Maks IOPS	80,000 [3] (25.6 KiB I/O [4)]	16,000 (16 KiB I/O [4])	256.000 [3] (16 KiB I/O [4])	64.000 (16 KiB I/O [4])
Throughput maksimal	2.000 MiB/s	250 MiB/s [1]	4000 MiB/s	1000 MiB/s [2]
Amazon EBS Multi-attach	Tidak didukung		Didukung	

	Volume SSD Tujuan Umum Amazon EBS	Volume SSD IOPS yang disediakan Amazon EBS	
Reservasi NVMe	Tidak didukung	Didukung	Didukung
Volume boot	Didukung		

[1] Batas throughput adalah antara 128 MiB/s dan 250 MiB/s, tergantung pada ukuran volume. Untuk informasi selengkapnya, lihat `kinerja` volume gp2. Volume yang dibuat sebelum 3 Desember 2018 yang belum dimodifikasi sejak pembuatan mungkin tidak mencapai performa penuh kecuali Anda mengubah volume.

[2] Untuk mencapai throughput maksimum 1.000 MiB/s, volume harus disediakan dengan 64.000 IOPS dan harus dilampirkan ke instance. Nitro-based Volume yang dibuat sebelum 6 Desember 2017 yang belum dimodifikasi sejak pembuatan mungkin tidak mencapai performa penuh kecuali Anda memodifikasi volumenya.

[3] Nitro-based inst ans mendukung volume yang disediakan hingga 256.000 IOPS. Tipe instans lainnya dapat dilampirkan ke volume yang disediakan hingga 64.000 IOPS, tetapi dapat mencapai hingga 32.000 IOPS.

[4] Merupakan I/O ukuran yang diperlukan untuk mencapai IOPS maksimum dalam batas throughput volume.

[Vol] `io2` ume 5 Block Express dirancang untuk memberikan latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O .

[6] Di Outpost, volume gp3 mendukung ukuran hingga 16 TiB, IOPS hingga 16.000, dan throughput hingga 1.000. MiB/s

Untuk informasi selengkapnya tentang jenis SSD-backed volume, lihat berikut ini:

- Volume SSD Tujuan Umum Amazon EBS
- Volume SSD IOPS yang disediakan Amazon EBS

Volume hard disk drive (HDD)

HDD-backed volume dioptimalkan untuk beban kerja streaming besar di mana atribut kinerja dominan adalah throughput. Tipe volume HDD termasuk HDD dengan Throughput Dioptimalkan dan HDD Dingin Berikut ini adalah ringkasan kasus penggunaan dan karakteristik HDD-backed volume.

	Volume HDD Throughput Dioptimalkan	Volume Cold HDD
Tipe volume	`st1`	`sc1`
Daya tahan	Daya tahan 99,8% - 99,9% (tingkat kegagalan tahunan 0,1% - 0,2%)	
Kasus penggunaan	• Big data • Gudang data • Pemrosesan log	• Throughput-oriented penyimpanan untuk data yang jarang diakses • Skenario di mana biaya penyimpanan terendah adalah penting
Ukuran volume	125 GiB - 16 TiB	
IOPS maksimum per volume (1 Mi I/O B)	500	250
Throughput maksimal per volume	500 MiB/s	250 MiB/s
Amazon EBS Multi-attach	Tidak didukung	
Volume boot	Tidak didukung	

Untuk informasi selengkapnya tentang volume Hard disk drive (HDD), lihat Amazon EBS Throughput Volume HDD dan Cold HDD yang Dioptimalkan.

Volume generasi sebelumnya

Volume magnetik (`standard`) adalah volume generasi sebelumnya yang didukung oleh drive magnetik. Mereka cocok untuk beban kerja dengan set data kecil di mana data jarang diakses dan performanya bukan merupakan hal yang penting. Volume ini menghasilkan sekitar 100 IOPS secara rata-rata, dengan kapasitas lonjakan hingga ratusan IOPS, dan ukurannya dapat berkisar antara 1 GiB hingga 1 TiB.

> **Tip**
>
> Magnetik adalah tipe volume generasi sebelumnya. Jika Anda membutuhkan kinerja atau konsistensi kinerja yang lebih tinggi daripada volume generasi sebelumnya, sebaiknya gunakan salah satu jenis volume generasi saat ini.

Tabel berikut menjelaskan tipe volume EBS generasi sebelumnya.

	Magnetik
Tipe volume	`standard`
Kasus penggunaan	Beban kerja di mana data jarang diakses
Ukuran volume	1 GiB-1 TiB
Maks IOPS per volume	40–200
Throughput maksimal per volume	40—90 MiB/s
Volume boot	Didukung

Volume SSD Tujuan Umum Amazon EBS

Volume SSD Tujuan Umum (gp2 dan gp3) didukung oleh solid-state drive (SSD). Harga dan performa diseimbangkan untuk berbagai macam beban kerja transaksional. Ini termasuk desktop virtual, basis data instans tunggal berukuran sedang, aplikasi interaktif sensitif latensi, lingkungan pengembangan dan pengujian, dan volume boot. Kami merekomendasikan volume ini untuk sebagian besar beban kerja.

Amazon EBS menawarkan tipe volume SSD Tujuan Umum berikut:

Tipe

- Volume SSD Tujuan Umum (gp3)
- Volume SSD Tujuan Umum (gp2)

Volume SSD Tujuan Umum (gp3)

Volume SSD Tujuan Umum (gp3) adalah generasi terbaru dari volume SSD Tujuan Umum, dan volume SSD dengan biaya terendah yang ditawarkan oleh Amazon EBS. Tipe volume ini membantu memberikan keseimbangan harga dan performa yang tepat untuk sebagian besar aplikasi. Ini juga membantu Anda menskalakan performa volume secara independen dari ukuran volume. Ini berarti Anda dapat menyediakan performa yang diperlukan tanpa perlu menyediakan kapasitas penyimpanan blok tambahan. Selain itu, volume gp3 menawarkan harga 20 persen lebih rendah per GiB daripada volume SSD Tujuan Umum (gp2).

Volume gp3 memberikan latensi milidetik satu digit dan daya tahan volume 99,8 persen hingga 99,9 persen dengan tingkat kegagalan tahunan (AFR) tidak lebih tinggi dari 0,2 persen, yang berarti maksimum dua kegagalan volume per 1.000 volume berjalan selama periode satu tahun. AWS mendesain volume gp3 untuk memberikan kinerja yang disediakan 99 persen dari waktu.

> **Tip**
>
> Untuk beban kerja yang sensitif terhadap latensi, sebaiknya gunakan volume io2 Block Express. `io2` Volume Block Express dirancang untuk memberikan latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O . `io2` Volume Block Express juga memberikan latensi outlier yang lebih baik dibandingkan dengan volume Tujuan Umum, mengurangi frekuensi I/Os melebihi 800 mikrodetik lebih dari 10 kali lipat. Untuk informasi selengkapnya, lihat Volume Blok Express IOPS SSD (`io2`) yang disediakan.

Daftar Isi

- Performa volume gp3
- Ukuran volume gp3
- Migrasi ke gp3 dari gp2

Performa volume gp3

> **Tip**
>
> Volume gp3 tidak menggunakan performa lonjakan. Volume ini dapat mempertahankan performa IOPS yang tersedia dan throughput secara penuh tanpa batas waktu.

Performa IOPS

Volume gp3 memberikan performa IOPS dasar yang konsisten 3.000 IOPS, yang disertakan dengan harga penyimpanan. Anda dapat menyediakan IOPS tambahan (hingga maksimum 80.000) dengan biaya tambahan dengan rasio 500 IOPS per GiB ukuran volume. IOPS maksimum dapat disediakan untuk volume 160 GiB atau lebih besar (500 IOPS per GiB × 160 GiB = 80.000 IOPS).

Performa throughput

Volume gp3 memberikan kinerja throughput dasar yang konsisten sebesar 125 MiB/s, yang disertakan dengan harga penyimpanan. Anda dapat menyediakan throughput tambahan (hingga maksimum 2.000 MiB/s) dengan biaya tambahan dengan rasio 0,25 MiB/s per IOPS yang disediakan. Throughput maksimum dapat disediakan pada 8.000 IOPS atau lebih tinggi dan 16 GiB atau lebih besar (8.000 IOPS × 0,25 MiB/s per IOPS = 2.000). MiB/s

> **Note**
>
> Di Outpost, volume gp3 mendukung ukuran hingga 16 TiB, IOPS hingga 16.000, dan throughput hingga 1.000. MiB/s

Ukuran volume gp3

Volume gp3 dapat berkisar dalam ukuran dari 1 GiB hingga 64 TiB.

Migrasi ke gp3 dari gp2

Jika saat ini Anda menggunakan volume gp2, Anda dapat memigrasikan volume ke gp3 menggunakan operasi Memodifikasi volume Amazon EBS menggunakan operasi Volume Elastic. Anda dapat menggunakan operasi Volume Elastis Amazon EBS untuk mengubah tipe volume, IOPS, dan throughput volume yang ada tanpa mengganggu instans Amazon EC2. Saat menggunakan

konsol untuk membuat volume atau membuat AMI dari snapshot, SSD Tujuan Umum gp3 adalah pilihan default untuk tipe volume. Dalam kasus lain, gp2 adalah pilihan default. Dalam kasus ini, Anda dapat memilih gp3 sebagai tipe volume alih-alih menggunakan gp2.

Untuk mengetahui berapa banyak yang dapat Anda hemat dengan memigrasikan volume gp2 Anda ke gp3, gunakan kalkulator penghematan biaya migrasi Amazon EBS gp2 ke gp3.

Volume SSD Tujuan Umum (gp2)

Volume SSD Tujuan Umum menawarkan penyimpanan hemat biaya yang ideal untuk berbagai beban kerja. Dengan volume gp2, performa diskalakan dengan ukuran volume.

> **Tip**
>
> Volume gp3 adalah generasi terbaru dari volume SSD Tujuan Umum. Volume itu menawarkan penskalaan performa yang lebih dapat diprediksi dan harga yang lebih murah hingga 20 persen daripada volume gp2. Untuk informasi selengkapnya, lihat Volume SSD Tujuan Umum (gp3).
> Untuk mengetahui berapa banyak yang dapat Anda hemat dengan memigrasikan gp2 volume ke gp3, gunakan kalkulator penghematan biaya migrasi Amazon EBS gp2 ke gp3.

gp2volume memberikan latensi milidetik satu digit dan daya tahan volume 99,8 persen hingga 99,9 persen dengan tingkat kegagalan tahunan (AFR) tidak lebih tinggi dari 0,2 persen, yang berarti maksimum dua kegagalan volume per 1.000 volume yang berjalan selama periode satu tahun. AWS mendesain gp2 volume untuk memberikan kinerja yang disediakan 99 persen dari waktu.

Daftar Isi

- kinerja volume gp2
- ukuran volume gp2

kinerja volume gp2

Performa IOPS

Skala performa IOPS dasar secara linear antara minimal 100 dan maksimum 16.000 dengan kecepatan 3 IOPS per GiB ukuran volume. Performa IOPS disediakan sebagai berikut:

- Volume 33,33 GiB dan yang lebih kecil disediakan dengan minimal 100 IOPS.

- Volume lebih besar dari 33,33 GiB disediakan dengan 3 IOPS per GiB ukuran volume hingga batas 16.000 IOPS, yang dicapai pada 5.334 GiB (3 X 5.334).
- Volume 5.334 GiB dan lebih besar disediakan dengan 16.000 IOPS.

Volume gp2 yang lebih kecil dari 1 TiB (dan yang disediakan dengan kurang dari 3.000 IOPS) dapat melonjak menjadi 3.000 IOPS bila diperlukan untuk jangka waktu yang lama. Kemampuan volume untuk meledak diatur oleh I/O kredit. Ketika I/O permintaan lebih besar dari kinerja dasar, volume menghabiskan I/O kredit untuk meledak ke tingkat kinerja yang diperlukan (hingga 3.000 IOPS). Saat meledak, I/O kredit tidak diakumulasikan dan dihabiskan pada tingkat IOPS yang digunakan di atas IOPS dasar (tingkat pengeluaran = burst IOPS - IOPS dasar). Semakin banyak I/O kredit yang diperoleh volume, semakin lama ia dapat mempertahankan kinerja ledakannya. Anda dapat menghitung Durasi lonjakan sebagai berikut:

```
                      (I/O credit balance)
Burst duration  =  ------------------------------
                  (Burst IOPS) - (Baseline IOPS)
```

Ketika I/O permintaan turun ke tingkat kinerja dasar atau lebih rendah, volume mulai mendapatkan I/O kredit pada tingkat 3 I/O kredit per GiB ukuran volume per detik. Volume memiliki batas akrual I/O kredit sebesar 5,4 juta I/O kredit, yang cukup untuk mempertahankan kinerja burst maksimum 3.000 IOPS selama setidaknya 30 menit.

> **Note**
>
> Setiap volume menerima saldo I/O kredit awal sebesar 5,4 juta I/O kredit, yang menyediakan siklus boot awal yang cepat untuk volume boot dan pengalaman bootstrap yang baik untuk aplikasi lain.

Tabel berikut mencantumkan contoh ukuran volume dan kinerja dasar terkait volume, durasi burst (saat dimulai dengan 5,4 juta I/O kredit), dan waktu yang diperlukan untuk mengisi ulang saldo I/O kredit kosong.

Ukuran volume (GiB)	Performa dasar (IOPS)	Durasi lonjakan pada 3.000 IOPS (detik)	Waktu untuk mengisi ulang saldo kredit kosong (detik)
1 hingga 33,33	100	1.862	54,000
100	300	2.000	18.000
334 (Ukuran minimum untuk throughput maksimal)	1.002	2.703	5.389
750	2.250	7.200	2.400
1.000	3000	N/A*	N/A*
5.334 (ukuran minimum untuk IOPS maks) dan lebih besar	16.000	N/A*	N/A*

* Performa dasar volume melebihi performa lonjakan maksimum.

Anda dapat memantau saldo I/O kredit untuk volume menggunakan `BurstBalance` metrik Amazon EBS di Amazon CloudWatch. Metrik ini menunjukkan persentase I/O kredit untuk gp2 sisa. Untuk informasi selengkapnya, lihat Karakter I/O istik dan pemantauan Amazon EBS. Anda dapat mengatur alarm yang memberi tahu Anda kapan nilai `BurstBalance` turun ke tingkat tertentu. Untuk informasi selengkapnya, lihat Membuat CloudWatch Alarm.

Performa throughput

gp2volume menghasilkan throughput antara 128 MiB/s dan 250 MiB/s, tergantung pada ukuran volume. Performa throughput disediakan sebagai berikut:

- Volume 170 GiB dan lebih kecil menghasilkan throughput maksimum 128 MiB/s.
- Volume yang lebih besar dari 170 GiB tetapi lebih kecil dari 334 GiB dapat meledak hingga throughput maksimum 250. MiB/s
- Volume yang 334 GiB dan lebih besar menghasilkan 250 MiB/s.

Throughput untuk gp2 volume dapat dihitung menggunakan rumus berikut, hingga batas throughput 250 MiB/s:

```
Throughput in MiB/s = IOPS performance × I/O size in KiB / 1,024
```

ukuran volume gp2

Volume gp2 dapat berkisar dalam ukuran dari 1 GiB hingga 16 TiB. Perlu diingat bahwa performa volume menskalakan secara linier dengan ukuran volume.

Volume SSD IOPS yang disediakan Amazon EBS

Volume SSD IOPS yang tersedia didukung oleh solid-state drive (SSD). Ini adalah volume penyimpanan Amazon EBS berkinerja tertinggi yang dirancang untuk beban kerja kritis IOPS-intensive, dan intensif throughput yang memerlukan latensi rendah. Volume SSD IOPS yang tersedia memberikan performa IOPS yang tersedia 99,9 persen waktu.

Amazon EBS menawarkan dua tipe volume SSD IOPS yang tersedia:

- Volume Blok Express IOPS SSD (io2) yang disediakan
- Volume IOPS SSD (io 1) yang disediakan

Volume Blok Express IOPS SSD (**io2**) yang disediakan

Volume Block Express io2 dibangun pada penyimpanan server arsitektur Amazon EBS generasi berikutnya. Ini telah dibangun untuk tujuan memenuhi persyaratan kinerja aplikasi I/O intensif yang paling menuntut yang berjalan pada instans yang dibangun pada Sistem Nitro. Dengan daya tahan tertinggi dan latensi terendah, Block Express sangat ideal untuk menjalankan beban kerja yang intensif performa, misi kritis, seperti Oracle, SAP HANA, Microsoft SQL Server, dan SAS Analytics.

Arsitektur Block Express meningkatkan performa dan skala volume io2. Server Block Express berkomunikasi dengan Nitro-based instans menggunakan protokol jaringan Scalable Reliable Datagram (SRD). Antarmuka ini diimplementasikan dalam Kartu Nitro yang didedikasikan untuk I/O fungsi Amazon EBS pada perangkat keras host instans. Ini meminimalkan variasi I/O penundaan dan latensi (jitter jaringan), yang memberikan kinerja yang lebih cepat dan lebih konsisten untuk aplikasi Anda.

Volume io2 Block Express dirancang untuk memberikan 99,999 persen daya tahan volume dengan tingkat kegagalan tahunan (AFR) tidak lebih dari 0,001 persen, yang berarti satu kegagalan volume

per 100.000 volume berjalan selama periode satu tahun. io2 Volume Block Express cocok untuk beban kerja yang mendapat manfaat dari satu volume yang memberikan latensi sub-milidetik yang konsisten, dan mendukung IOPS dan throughput yang lebih tinggi daripada volume gp3.

Saat dilampirkan ke Nitro-based inst io2 ans, volume Block Express dirancang untuk memberikan latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O . io2 Volume Block Express juga memberikan latensi outlier yang lebih baik dibandingkan dengan volume Tujuan Umum, mengurangi frekuensi I/Os melebihi 800 mikrodetik lebih dari 10 kali lipat.

Topik

- Pertimbangan-pertimbangan
- Performa

Pertimbangan-pertimbangan

- io2Volume Block Express tersedia di semua Wil AWS ayah, termasuk Wil AWS GovCloud (US) ayah dan Wilayah China.
- Pada 30 April 2025, semua volume baru dan yang dibuat sebelumnya adalah io2 volume io2 Block Express.
- Nitro-based inst ans mendukung volume yang disediakan hingga 256.000 IOPS. Tipe instans lainnya dapat dilampirkan ke volume yang disediakan hingga 64.000 IOPS, tetapi dapat mencapai hingga 32.000 IOPS.

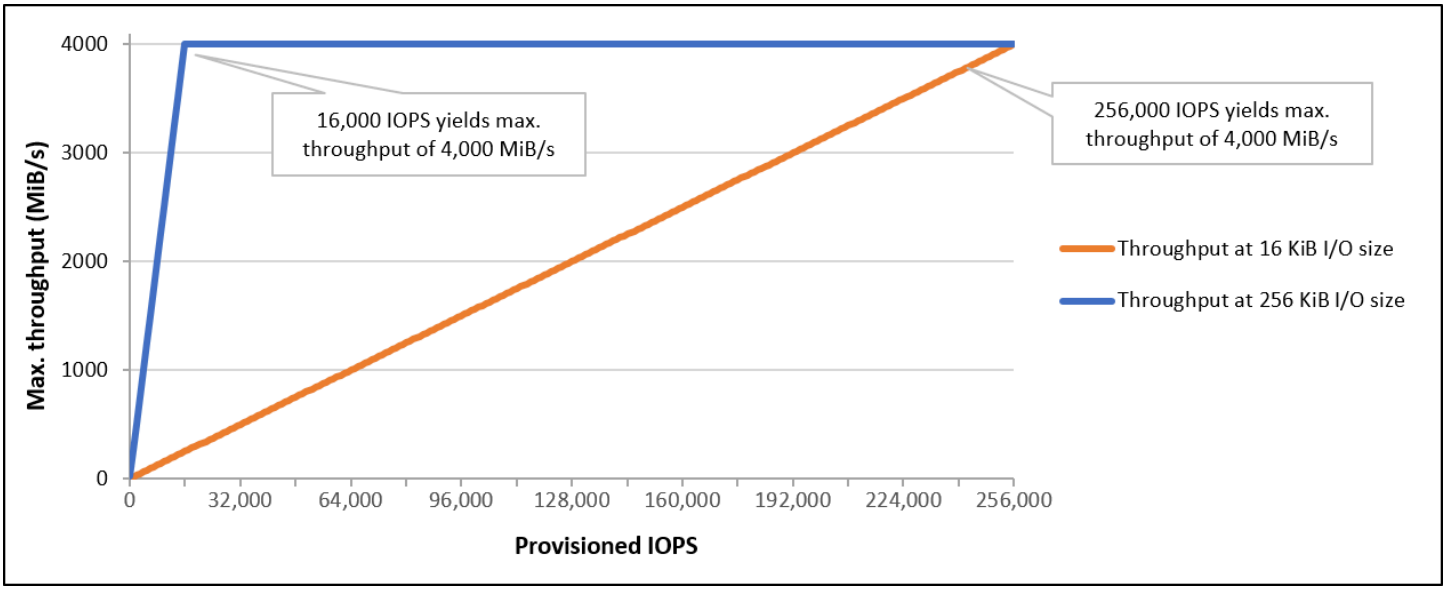
Performa

io2Volume Block Express memiliki karakteristik sebagai berikut:

- Latensi rata-rata di bawah 500 mikrodetik untuk ukuran 16KiB I/O . Latensi outlier yang lebih baik dibandingkan dengan volume Tujuan Umum, mengurangi frekuensi I/Os melebihi 800 mikrodetik lebih dari 10 kali lipat.
- Kapasitas penyimpanan hingga 64 TiB (65.536 GiB).
- IOPS yang disediakan hingga 256.000, dengan IOPS:GiB rasio 1.000:1. IOPS maksimum dapat disediakan dengan volume 256 GiB dan lebih besar (1.000 IOPS × 256 GiB = 256.000 IOPS).

> **Note**
>
> Anda dapat mencapai hingga 256.000 IOPS dengan Nitro-based instans. Pada contoh lain, Anda dapat mencapai hingga 32.000 IOPS.

- Volume throughput hingga 4.000 MiB/s. Throughput diskalakan secara proporsional pada tingkat 0,256 MiB/s per IOPS yang disediakan. Throughput maksimum dapat dicapai pada 16.000 IOPS atau lebih tinggi.



Volume IOPS SSD (**io** 1) yang disediakan

Volume IOPS SSD (`io1`) yang disediakan dirancang untuk memenuhi kebutuhan beban kerja, I/O-intensive khususnya beban kerja database, yang sensitif terhadap kinerja dan konsistensi penyimpanan. Volume SSD IOPS yang tersedia menggunakan tingkat IOPS yang konsisten, yang Anda tentukan saat membuat volume, dan Amazon EBS memberikan performa yang telah disediakan sebesar 99,9 persen.

`io1` volume dirancang untuk memberikan 99,8 hingga 99,9 persen daya tahan volume dengan tingkat kegagalan tahunan (AFR) tidak lebih dari 0,2 persen, yang berarti maksimum dua kegagalan volume per 1.000 volume berjalan selama periode satu tahun.

Volume `io1` tersedia untuk semua tipe instans Amazon EC2.

Kinerja

Ukuran volume `io1` dapat berkisar dari 4 GiB hingga 16 TiB dan Anda dapat menyediakan dari 100 IOPS hingga 64.000 IOPS per volume. Rasio maksimum IOPS yang tersedia dengan ukuran volume yang diminta (dalam GiB) adalah 50:1. Misalnya, volume `io1` 100 GiB dapat disediakan hingga 5.000 IOPS.

IOPS maksimum dapat ditetapkan untuk volume yang 1.280 GiB atau lebih besar (50 × 1.280 GiB = 64.000 IOPS).

- `io1`volume yang disediakan hingga 32.000 IOPS mendukung I/O ukuran maksimum 256 KiB dan menghasilkan sebanyak 500 MiB/s throughput. Dengan I/O ukuran maksimum, throughput puncak dicapai pada 2.000 IOPS.
- Volume `io1` yang disediakan dengan lebih dari 32.000 IOPS (hingga maksimal 64.000 IOPS) menghasilkan peningkatan throughput linier pada tingkat 16 KiB per IOPS yang tersedia. Misalnya, volume yang disediakan dengan 48.000 IOPS dapat mendukung hingga 750 MiB/s throughput (16 KiB per IOPS yang disediakan × 48.000 IOPS yang disediakan = 750). MiB/s
- Untuk mencapai throughput maksimum 1.000 MiB/s, volume harus disediakan dengan 64.000 IOPS (16 KiB per IOPS yang disediakan × 64.000 IOPS yang disediakan = 1.000). MiB/s
- Anda dapat mencapai hingga 64.000 IOPS hanya pada Nitro-based instans. Pada contoh lain, Anda dapat mencapai hingga 32.000 IOPS.

. Grafik berikut menggambarkan karakteristik performa ini:



Pengalaman per I/O latensi Anda bergantung pada IOPS yang disediakan dan profil beban kerja Anda. Untuk pengalaman I/O latensi terbaik, pastikan Anda menyediakan IOPS untuk memenuhi I/O profil beban kerja Anda.

Amazon EBS Throughput Volume HDD dan Cold HDD yang Dioptimalkan

Vol HDD-backed ume yang disediakan oleh Amazon EBS termasuk dalam kategori ini:

- HDD Throughput Dioptimalkan — HDD hemat biaya yang dirancang untuk beban kerja yang sering diakses dan membutuhkan banyak throughput.
- Cold HDD — HDD dengan biaya terendah yang dirancang untuk beban kerja yang jarang diakses.

Topik

- Pembatasan pada throughput per-instans
- Volume HDD Throughput Dioptimalkan
- Volume Cold HDD
- Pertimbangan performa saat menggunakan volume HDD
- Pantau saldo bucket lonjakan untuk volume

Pembatasan pada throughput per-instans

Throughput untuk volume `st1` dan `sc1` selalu ditentukan oleh yang lebih kecil berikut ini:

- Batas throughput volume
- Batas throughput instans

Untuk semua volume Amazon EBS, kami sarankan Anda memilih instans EBS-optimized EC2 yang sesuai untuk menghindari kemacetan jaringan.

Volume HDD Throughput Dioptimalkan

Volume HDD Throughput yang Dioptimalkan (`st1`) menyediakan penyimpanan magnetik hemat biaya yang mendefinisikan performa dalam hal throighput daripada IOPS. Tipe volume ini cocok untuk beban kerja yang besar dan berurutan seperti Amazon EMR, ETL, gudang data, dan pemrosesan log. Volume `st1` yang dapat di-boot tidak didukung.

Volume HDD Throughput Dioptimalkan (`st1`), meskipun serupa dengan volume Cold HDD (`sc1`), dirancang untuk mendukung data yang sering diakses.

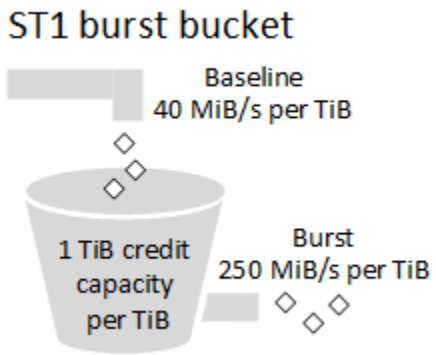
> **Note**
>
> Jenis volume ini dioptimalkan untuk beban kerja yang melibatkan beban kerja besar, berurutan I/O, dan kami menyarankan agar pelanggan dengan beban kerja melakukan I/O penggunaan Volume SSD Tujuan Umum Amazon EBS kecil, acak atau. Volume SSD IOPS yang disediakan Amazon EBS Untuk informasi selengkapnya, lihat Inefisiensi kecil read/ writes pada HDD.

Volume Throughput Optimized HDD (`st1`) yang terpasang pada EBS-optimized instans dirancang untuk menawarkan kinerja yang konsisten, memberikan setidaknya 90 persen dari kinerja throughput yang diharapkan 99 persen dari waktu pada tahun tertentu.

Kredit throughput dan performa lonjakan

Seperti `gp2`, `st1` menggunakan model bucket lonjakan untuk performa. Ukuran volume menentukan throughput tingkat dasar volume Anda, yang merupakan tingkat di mana volume mengakumulasi kredit throughput. Ukuran volume juga menentukan throughput lonjakan volume Anda, yang merupakan tingkat di mana Anda dapat menghabiskan kredit saat tersedia. Volume yang lebih besar memiliki garis dasar dan throughput lonjakan yang lebih tinggi. Semakin banyak kredit yang dimiliki volume Anda, semakin lama ia dapat melaju I/O pada level burst.

Diagram berikut menunjukkan perilaku bucket lonjakan untuk `st1`.



Tergantung pada batas throughput dan kredit throughput, throughput volume `st1` dinyatakan dengan rumus berikut:

```
(Volume size) × (Credit accumulation rate per TiB) = Throughput
```

Untuk `st1` volume 1-TiB, throughput burst dibatasi hingga 250 MiB/s, bucket diisi dengan kredit pada 40 MiB/s, dan dapat menampung hingga 1 TiB-worth kredit.

Volume yang lebih besar menskalakan batas-batas ini secara linier, dengan throughput dibatasi maksimal 500. MiB/s Setelah bucket habis, throughput dibatasi pada tingkat dasar 40 MiB/s per TiB.

Pada ukuran volume mulai dari 0,125 TiB hingga 16 TiB, throughput baseline bervariasi dari 5 MiB/s hingga batas 500 MiB/s, yang dicapai pada 12,5 TiB sebagai berikut:

```
            40 MiB/s
12.5 TiB × ---------- = 500 MiB/s
             1 TiB
```

Throughput burst bervariasi dari 31 MiB/s hingga batas 500 MiB/s, yang dicapai pada 2 TiB sebagai berikut:

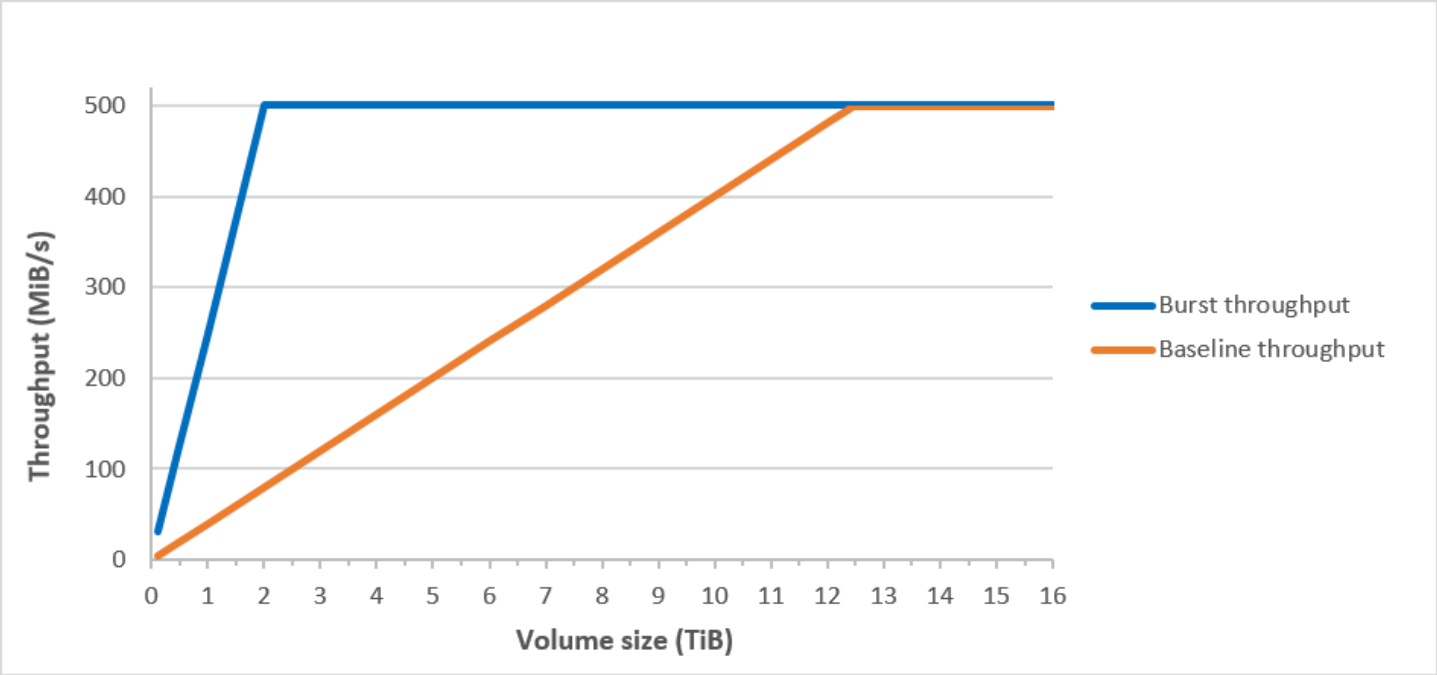
```
         250 MiB/s
2 TiB × ---------- = 500 MiB/s
          1 TiB
```

Tabel berikut ini menyatakan rentang lengkap nilai throughput dasar dan lonjakan untuk `st1`.

Ukuran volume (TiB)	Throughput dasar ST1 () MiB/s	Throughput burst ST1 () MiB/s
0.125	5	31
0.5	20	125
1	40	250
2	80	500
3	120	500
4	160	500
5	200	500
6	240	500
7	280	500
8	320	500

Ukuran volume (TiB)	Throughput dasar ST1 () MiB/s	Throughput burst ST1 () MiB/s
9	360	500
10	400	500
11	440	500
12	480	500
12.5	500	500
13	500	500
14	500	500
15	500	500
16	500	500

Diagram berikut membuat plot nilai tabel:



> **Note**
>
> Saat Anda membuat snapshot dari volume HDD Throughput Dioptimalkan (`st1`), performa dapat menurun sejauh nilai dasar volume saat snapshot sedang berlangsung.

Untuk informasi tentang menggunakan CloudWatch metrik dan alarm untuk memantau saldo bucket burst Anda, lihatPantau saldo bucket lonjakan untuk volume.

Volume Cold HDD

Volume Cold HDD (`sc1`) menyediakan penyimpanan magnetik hemat biaya yang mendefinisikan performa dalam hal throughput daripada IOPS. Dengan batas throughput yang lebih rendah dari `st1`, `sc1` cocok untuk beban kerja cold-data yang besar dan berurutan. Jika Anda memerlukan akses yang jarang ke data Anda dan ingin menghemat biaya, `sc1` menyediakan penyimpanan blok murah. Volume `sc1` yang dapat di-boot tidak didukung.

Volume Cold HDD (`sc1`), meskipun serupa dengan volume HDD Throughput Dioptimalkan (`st1`), dirancang untuk mendukung data yang jarang diakses.

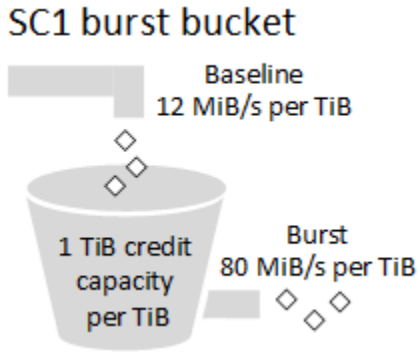
> **Note**
>
> Jenis volume ini dioptimalkan untuk beban kerja yang melibatkan beban kerja besar, berurutan I/O, dan kami menyarankan agar pelanggan dengan beban kerja melakukan I/O penggunaan Volume SSD Tujuan Umum Amazon EBS kecil, acak atau. Volume SSD IOPS yang disediakan Amazon EBS Untuk informasi selengkapnya, lihat Inefisiensi kecil read/ writes pada HDD.

Volume Cold HDD (`sc1`) yang terpasang pada EBS-optimized instans dirancang untuk menawarkan kinerja yang konsisten, memberikan setidaknya 90 persen dari kinerja throughput yang diharapkan 99 persen dari waktu pada tahun tertentu.

Kredit throughput dan performa lonjakan

Seperti gp2, `sc1` menggunakan model bucket lonjakan untuk performa. Ukuran volume menentukan throughput tingkat dasar volume Anda, yang merupakan tingkat di mana volume mengakumulasi kredit throughput. Ukuran volume juga menentukan throughput lonjakan volume Anda, yang

merupakan tingkat di mana Anda dapat menghabiskan kredit saat tersedia. Volume yang lebih besar memiliki garis dasar dan throughput lonjakan yang lebih tinggi. Semakin banyak kredit yang dimiliki volume Anda, semakin lama ia dapat melaju I/O pada level burst.



Tergantung pada batas throughput dan kredit throughput, throughput volume `sc1` dinyatakan dengan rumus berikut:

```
(Volume size) × (Credit accumulation rate per TiB) = Throughput
```

Untuk `sc1` volume 1-TiB, throughput burst dibatasi hingga 80 MiB/s, bucket diisi dengan kredit pada 12 MiB/s, dan dapat menampung hingga 1 TiB-worth kredit.

Volume yang lebih besar menskalakan batas-batas ini secara linier, dengan throughput dibatasi maksimal 250. MiB/s Setelah bucket habis, throughput dibatasi pada tingkat dasar 12 MiB/s per TiB.

Pada ukuran volume mulai dari 0,125 TiB hingga 16 TiB, throughput baseline bervariasi dari 1,5 MiB/s hingga maksimum 192 MiB/s, yang dicapai pada 16 TiB sebagai berikut:

```
           12 MiB/s
16 TiB × ---------- = 192 MiB/s
            1 TiB
```

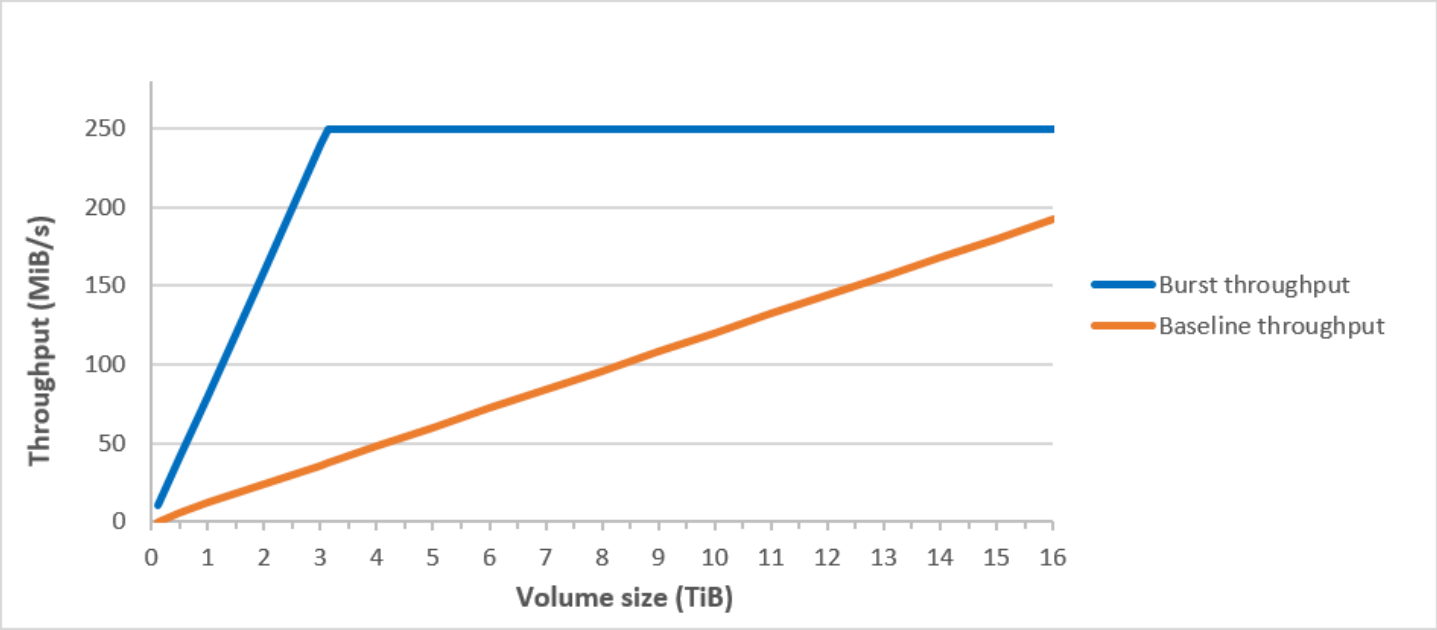
Throughput burst bervariasi dari 10 MiB/s hingga batas 250 MiB/s, yang dicapai pada 3,125 TiB sebagai berikut:

```
             80 MiB/s
3.125 TiB × ----------- = 250 MiB/s
              1 TiB
```

Tabel berikut ini menyatakan rentang lengkap nilai throughput dasar dan lonjakan untuk `sc1`:

Ukuran Volume (TiB)	Throughput Dasar SC1 () MiB/s	Throughput Burst SC1 () MiB/s
0.125	1.5	10
0.5	6	40
1	12	80
2	24	160
3	36	240
3.125	37,5	250
4	48	250
5	60	250
6	72	250
7	84	250
8	96	250
9	108	250
10	120	250
11	132	250
12	144	250
13	156	250
14	168	250
15	180	250
16	192	250

Diagram berikut membuat plot nilai tabel:



> **Note**
>
> Saat Anda membuat snapshot volume Cold HDD (`sc1`), performa dapat menurun sejauh nilai dasar volume saat snapshot sedang berlangsung.

Untuk informasi tentang menggunakan CloudWatch metrik dan alarm untuk memantau saldo bucket burst Anda, lihatPantau saldo bucket lonjakan untuk volume.

Pertimbangan performa saat menggunakan volume HDD

Untuk hasil throughput optimal yang menggunakan volume HDD, rencanakan beban kerja Anda dengan mempertimbangkan hal-hal berikut.

Membandingkan HDD Throughput Dioptimalkan dan Cold HDD

Ukuran bucket `st1` dan `sc1` bervariasi sesuai dengan ukuran volume, dan bucket penuh berisi token yang cukup untuk pemindaian volume penuh. Namun, volume `st1` dan `sc1` yang lebih besar membutuhkan waktu yang lebih lama untuk menyelesaikan pemindaian volume karena batas throughput per-instans dan per-volume. Volume yang terpasang pada instans yang lebih kecil dibatasi berdasarkan pada throughput per instans daripada batas throughput `st1` atau `sc1`.

K `st1` edu `sc1` anya dan dirancang untuk konsistensi kinerja 90 persen dari throughput burst 99 persen dari waktu. Non-compliant periode kira-kira didistribusikan secara seragam, menargetkan 99 persen dari total throughput yang diharapkan setiap jam.

Secara umum, waktu pemindaian dinyatakan dengan rumus ini:

```
 Volume size
------------ = Scan time
 Throughput
```

Misalnya, mempertimbangkan jaminan konsistensi performa dan optimisasi lainnya, pelanggan `st1` dengan volume 5-TiB dapat melakukan pemindaian volume penuh dalam 2,91 hingga 3,27 jam.

- Waktu pemindaian optimal

```
   5 TiB            5 TiB
----------- = ------------------ = 10,486 seconds = 2.91 hours
 500 MiB/s     0.00047684 TiB/s
```

- Waktu pemindaian maksimal

```
  2.91 hours
-------------- = 3.27 hours
 (0.90)(0.99) <-- From expected performance of 90% of burst 99% of the time
```

Demikian pula, pelanggan `sc1` dengan volume 5TiB dapat melakukan pemindaian volume penuh dalam waktu 5,83 hingga 6,54 jam.

- Waktu pemindaian optimal

```
   5 TiB             5 TiB
----------- = ------------------- = 20972 seconds = 5.83 hours
 250 MiB/s     0.000238418 TiB/s
```

- Waktu pemindaian maksimal

```
  5.83 hours
-------------- = 6.54 hours
 (0.90)(0.99)
```

Tabel berikut menunjukkan waktu pemindaian ideal untuk volume berbagai ukuran, dengan asumsi bucket penuh dan throughput instans yang cukup.

Ukuran volume (TiB)	Waktu pemindaian ST1 dengan lonjakan (jam)*	Waktu pemindaian SC1 dengan lonjakan (jam)*
1	1.17	3.64
2	1.17	3.64
3	1.75	3.64
4	2.33	4.66
5	2.91	5.83
6	3.50	6.99
7	4.08	8.16
8	4.66	9.32
9	5.24	10.49
10	5.83	11.65
11	6.41	12.82
12	6.99	13.98
13	7.57	15.15
14	8.16	16.31
15	8.74	17.48
16	9.32	18.64

* Waktu pemindaian ini mengasumsikan kedalaman antrian rata-rata (dibulatkan ke bilangan bulat terdekat) empat atau lebih saat melakukan 1 MiB berurutan I/O.

Oleh karena itu, jika Anda memiliki beban kerja berorientasi throughput yang perlu menyelesaikan pemindaian dengan cepat (hingga 500 MiB/s), atau memerlukan beberapa pemindaian volume penuh sehari, gunakan. `st1` Jika Anda mengoptimalkan biaya, data Anda relatif jarang diakses, dan Anda tidak memerlukan lebih dari 250 kinerja MiB/s pemindaian, maka gunakan`sc1`.

Inefisiensi kecil read/writes pada HDD

Model kinerja untuk `st1` dan `sc1` volume dioptimalkan untuk berurutan I/Os, mendukung beban kerja throughput tinggi, menawarkan kinerja yang dapat diterima pada beban kerja dengan IOPS dan throughput campuran, dan mencegah beban kerja dengan kecil dan acak. I/O

Misalnya, I/O permintaan 1 MiB atau kurang dihitung sebagai kredit 1 MiB I/O . Namun, jika ber I/Os urutan, mereka digabungkan menjadi I/O blok 1 MiB dan hanya dihitung sebagai kredit 1 MiB I/O .

Pantau saldo bucket lonjakan untuk volume

Anda dapat memantau level bucket burst untuk `st1` dan `sc1` volume menggunakan `BurstBalance` metrik Amazon EBS yang tersedia di Amazon CloudWatch. Metrik ini menunjukkan kredit throughput untuk `st1` dan `sc1` yang tersisa di bucket lonjakan. Untuk informasi selengkapnya tentang `BurstBalance` metrik dan metrik lain yang terkait I/O, lihatKarakter I/O istik dan pemantauan Amazon EBS. CloudWatch juga memungkinkan Anda untuk mengatur alarm yang memberi tahu Anda ketika nil `BurstBalance` ainya jatuh ke tingkat tertentu. Untuk informasi selengkapnya, lihat Membuat CloudWatch Alarm.

Batasan volume Amazon EBS

Ukuran volume Amazon EBS dibatasi oleh fisika dan aritmatika penyimpanan data blok, serta oleh keputusan implementasi sistem operasi (OS) dan perancang sistem file. AWS memberlakukan batasan tambahan pada ukuran volume untuk menjaga keandalan layanannya.

Bagian-bagian berikut menjelaskan faktor terpenting yang membatasi ukuran volume EBS yang dapat digunakan dan menawarkan rekomendasi untuk mengonfigurasi volume EBS Anda.

Daftar Isi

- Kapasitas penyimpanan
- Pembatasan layanan
- Skema partisi

- Ukuran blok data

Kapasitas penyimpanan

Tabel berikut merangkum jadwal penyimpanan teoretis dan yang diimplementasikan untuk sistem file yang paling umum digunakan di Amazon EBS, dengan asumsi ukuran blok sebesar 4.096 bita.

Skema pembagiar	Blok maksimal yang dapat dihitung	Ukuran maks teoretis (blok × ukuran blok)	Ekst4 menerapkan ukuran maksimal*	XFS menerapkan ukuran maksimal**	NTFS menerapkan ukuran maksimal	Max yang didukung oleh EBS
MBR	2^{32}	2 TiB	2 TiB	2 TiB	2 TiB	2 TiB
GPT	2^{64}	64 ZiB	1 EiB = 1024^2 TiB (50 TiB disertifikasi pada RHEL7)	500 TiB (disertifikasi pada RHEL7)	256 TiB	64 TiB †

* Ext4 Howto dan Apa batas ukuran file dan sistem untuk Red Hat Enterprise Linux?

** Apa batas ukuran file dan sistem untuk Red Hat Enterprise Linux?

† Volume `io2` Block Express mendukung hingga 64 TiB untuk partisi GPT. Untuk informasi selengkapnya, lihat Volume Blok Express IOPS SSD (`io2`) yang disediakan.

Pembatasan layanan

Amazon EBS merupakan abstrak penyimpanan pusat data yang didistribusikan secara besar-besaran ke dalam hard disk virtual. Ke sistem operasi yang terpasang pada instans EC2 volume EBS terlampir tampaknya berupa hard disk fisik yang berisi sektor disk 512-bita. OS tersebut mengelola alokasi blok (atau klaster) data ke sektor virtual tersebut melalui pemanfaatan manajemen penyimpanan. Alokasi tersebut sesuai dengan skema partisi volume, seperti master boot record

(MBR) atau GUID partition table (GPT), dan sesuai kemampuan sistem file yang terpasang (ext4, NTFS, dan seterusnya).

EBS tidak mengetahui data yang terkandung di sektor disk virtual; tapi hanya memastikan integritas sektor. Ini berarti bahwa AWS tindakan dan tindakan OS tidak bergantung satu sama lain. Saat Anda memilih ukuran volume, perhatikan kemampuan dan batasan keduanya, seperti dalam kasus berikut:

- Saat ini EBS mendukung ukuran volume maksimum 64 TiB. Artinya, Anda dapat membuat volume EBS sebesar 64 TiB, tetapi apakah OS tersebut mengakui semua kapasitas itu tergantung pada karakteristik desainnya sendiri dan bagaimana volumenya dipartisi.
- Volume boot harus menggunakan skema partisi MBR atau GPT. AMI tempat Anda meluncurkan instance menentukan mode boot dan kemudian skema partisi yang digunakan untuk volume boot.

  Dengan MBR, volume boot dibatasi hingga ukuran 2 TiB.

  Dengan GPT, volume boot bisa berukuran hingga 64 TiB bila digunakan dengan GRUB2 (Linux) atau UEFI boot mode (Windows).

  Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.
- Non-boot volume yang 2 TiB (2048 GiB) atau lebih besar harus menggunakan tabel partisi GPT untuk mengakses seluruh volume.

Skema partisi

Di antara dampak lainnya, skema pembagian menentukan berapa banyak blok data logis yang dapat ditangani secara unik dalam satu volume. Untuk informasi selengkapnya, lihat Ukuran blok data. Skema partisi umum yang digunakan adalah Master Boot Record (MBR) dan tabel partisi GUID (GPT). Perbedaan penting antara skema ini dapat dirangkum sebagai berikut.

MBR

MBR menggunakan struktur data 32-bit untuk menyimpan alamat blok. Ini berarti bahwa setiap blok data dipetakan dengan salah satu dari 2^{32} bilangan bulat yang mungkin. Ukuran maksimum volume yang dapat dihitung diberikan dengan formula berikut ini:

$$2^{32} \times \text{Block size}$$

Ukuran blok untuk volume MBR secara konvensional dibatasi sebesar 512 bita. Oleh karena itu:

$$2^{32} \times 512 \text{ bytes} = 2 \text{ TiB}$$

Solusi teknik untuk meningkatkan batas 2-TiB ini untuk volume MBR belum memenuhi adopsi industri yang tersebar luas. Akibatnya, Linux dan Windows tidak pernah mendeteksi volume MBR lebih besar dari 2 TiB bahkan jika AWS menunjukkan ukurannya lebih besar.

GPT

GPT menggunakan struktur data 64-bit untuk menyimpan alamat blok. Ini berarti bahwa setiap blok data dipetakan dengan salah satu dari 2^{64} bilangan bulat yang mungkin. Ukuran maksimum volume yang dapat dihitung diberikan dengan formula berikut ini:

$$2^{64} \times \text{Block size}$$

Ukuran blok untuk volume GPT umumnya 4.096 bita. Oleh karena itu:

$$\begin{aligned} & 2^{64} \times 4{,}096 \text{ bytes} \\ &= 2^{64} \times 2^{12} \text{ bytes} \\ &= 2^{70} \times 2^{6} \text{ bytes} \\ &= 64 \text{ ZiB} \end{aligned}$$

Real-world sistem komputer tidak mendukung apa pun yang mendekati maksimum teoretis ini. Ukuran sistem file yang diterapkan saat ini dibatasi hingga 50 TiB untuk ext4 dan 256 TiB untuk NTFS.

Ukuran blok data

Penyimpanan data di hard drive modern dikelola melalui pengalamatan blok logis, lapisan abstraksi yang memungkinkan sistem operasi membaca dan menulis data dalam blok logis tanpa mengetahui banyak tentang perangkat keras yang mendasarinya. Sistem operasi bergantung pada perangkat penyimpanan untuk memetakan blok ke sektor fisiknya, dan membaca dan menulis data ke disk menggunakan blok data yang merupakan kelipatan dari ukuran sektor.

Amazon EBS mengiklankan sektor fisik 512 byte atau 4.096 byte (4 KiB) ke sistem operasi, tergantung pada faktor-faktor berikut:

1. Jenis instans Amazon EC2
2. Sistem operasi

3. Versi driver NVMe

Amazon EBS mengiklankan sektor fisik 4 KiB hanya jika semua faktor mendukungnya. Jika salah satu dari ini tidak mendukung sektor fisik 4 KiB, Amazon EBS mengiklankan sektor fisik 512 byte.

Dukungan jenis instans Amazon EC2

Tabel berikut menunjukkan ukuran sektor yang diiklankan Amazon EBS untuk berbagai jenis instans Amazon EC2.

Tipe instans	Linux	Windows
Semua jenis Xen-based instans	Amazon EBS selalu mengiklankan sektor fisik 512 byte	
A1 | C5 | C5a | C5ad | C5d | C5n | C6g | C6gD | DL1 | D3 | D3en | G4ad | G4dn | G5 | G5g | I3 | I3en | Inf1 | M5 | M5a | M5ad | M5d | M5dn | M5n | M5zn | M6g | M6g Gd | P3dN | P4d | P4de | R5 | R5a | R5ad | R5d | R5dn | R5n | R6g | R6gD | T3 | T3a | T4g | | | | | | X2Gd | X2ieZn | VT1 | Z1d U-12tb1 U-18tb1 U-24tb1 U-3tb1 U-6tb1 U-9tb1	Amazon EBS selalu mengiklankan sektor fisik 512 byte	Amazon EBS mengiklankan sektor fisik 512 byte atau 4 KiB [1]
Semua Nitro-based contoh lainnya	Amazon EBS mengiklankan sektor fisik 512 byte atau 4 KiB [1]	

[1] Tergantung pada dukungan sistem operasi. Lihat bagian berikut.

Dukungan sistem operasi

Tabel berikut memberikan contoh sistem operasi dan ukuran sektor fisik yang sesuai yang diiklankan oleh Amazon EBS. Ini bukan daftar lengkap. Kami menyarankan Anda memverifikasi ukuran sektor fisik yang diiklankan oleh Amazon EBS di sistem operasi Anda.

Sistem operasi	Ukuran sektor fisik yang diiklankan
• Amazon Linux dengan kernel versi 4.14 dan sebelumnya • RHEL 7.9 dan sebelumnya • Ubuntu 20.04 dan versi sebelumnya • Windows 7/Windows Server 2008 dan versi sebelumnya	512 byte
• Amazon Linux dengan kernel versi 5.3 dan yang lebih baru • RHEL8.8 dan kemudian • Ubuntu 22.04 dan yang lebih baru • Windows 8/Windows Server 2012 dan yang lebih baru [1]	4 KiB

[1] Untuk beban kerja Windows, pastikan Anda menggunakan driver AWS NVMe versi terbaru. Amazon EBS mengiklankan sektor fisik 4-KiB dengan driver AWS NVMe versi 1.4.1 dan yang lebih baru.

Non-default ukuran blok

Ukuran default industri untuk blok data logis saat ini adalah 4 KiB. Karena beban kerja tertentu mendapatkan keuntungan dari ukuran blok yang lebih kecil atau lebih besar, sistem file mendukung ukuran blok non-default yang dapat ditentukan selama pemformatan. Skenario di mana ukuran blok non-default harus digunakan (seperti pengoptimalan) berada di luar cakupan topik ini, tetapi pilihan ukuran blok memiliki konsekuensi untuk kapasitas penyimpanan volume. Tabel berikut menunjukkan kapasitas penyimpanan teoritis sebagai fungsi dari ukuran blok. Namun, perhatikan bahwa EBS-imposed batas ukuran volume (64 TiB untuk io2 Block Express) saat ini sama dengan ukuran maksimum yang diaktifkan oleh blok data 16 KiB.

Ukuran blok	Ukuran volume maksimal
4 KiB (default)	16 TiB

Ukuran blok	Ukuran volume maksimal
8 KiB	32 TiB
16 KiB	64 TiB
32 KiB	128 TiB
64 KiB (maksimal)	256 TiB

Volume dan NVMe Amazon EBS

Volume Amazon EBS diekspos sebagai perangkat blok NVMe pada instans Amazon EC2 yang dibangun di atas Sistem Nitro. AWS Untuk sepenuhnya memanfaatkan kinerja dan kemampuan volume Amazon EBS yang diekspos sebagai perangkat blok NVMe, instans EC2 harus menginstal driver AWS NVMe. Semua AMI AWS Windows dan Linux generasi saat ini dilengkapi dengan driver AWS NVMe yang diinstal secara default.

Jika Anda menggunakan AMI yang tidak memiliki driver AWS NVMe, Anda dapat menginstalnya secara manual. Untuk informasi selengkapnya, lihat driver AWS NVMe di Panduan Pengguna Amazon EC2.

Instans Linux

Nama perangkat adalah`/dev/nvme0n1`,`/dev/nvme1n1`, dan seterusnya. Nama perangkat yang Anda tentukan di pemetaan perangkat blok akan diganti 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.

Instans Windows

Saat Anda memasang volume ke instans, Anda menyertakan nama perangkat untuk volume tersebut. Nama perangkat ini digunakan oleh Amazon EC2 Driver perangkat blok untuk instance menetapkan nama volume aktual saat memasang volume, dan nama yang ditetapkan dapat berbeda dari nama yang digunakan Amazon EC2.

Daftar Isi

- Petakan volume Amazon EBS ke nama perangkat NVMe
- Waktu tunggu I/O operasi NVMe untuk volume Amazon EBS

- NVMe Batalkan perintah untuk volume Amazon EBS

Petakan volume Amazon EBS ke nama perangkat NVMe

EBS menggunakan I/O virtualisasi root tunggal (SR-IOV) untuk menyediakan lampiran volume pada Nitro-based instans yang menggunakan spesifikasi NVMe. Perangkat ini mengandalkan driver NVMe standar pada sistem operasi. Driver ini biasanya menemukan perangkat terpasang selama boot instans, dan membuat simpul perangkat berdasarkan urutan respons perangkat, bukan pada cara perangkat ditentukan dalam pemetaan perangkat blok.

Instans Linux

<y>Di Linux, nama perangkat NVMe mengikuti pola`/dev/nvme<x>n<y>`, di <x>mana urutan enumerasi, dan, untuk EBS, adalah 1. Terkadang, perangkat dapat merespons penemuan dalam urutan yang berbeda di awal instans berikutnya, yang menyebabkan nama perangkat berubah. Selain itu, nama perangkat yang ditetapkan oleh driver perangkat blok dapat berbeda dari nama yang ditentukan dalam pemetaan perangkat blok.

Kami menyarankan agar Anda menggunakan pengidentifikasi stabil untuk volume EBS dalam instans Anda, seperti salah satu dari berikut ini:

- Untuk Nitro-based contoh, pemetaan perangkat blok yang ditentukan di konsol Amazon EC2 saat Anda melampirkan volume EBS atau selama `AttachVolume` atau panggilan `RunInstances` API ditangkap di bidang data khusus vendor identifikasi pengontrol NVMe. Dengan AMI Amazon Linux setelah versi 2017.09.01, kami menyediakan aturan `udev` yang membaca data ini dan membuat tautan simbolis ke pemetaan perangkat blok.
- ID volume EBS dan titik pemasangan stabil di antara perubahan status instans. Nama perangkat NVMe dapat berubah tergantung pada urutan respons perangkat selama boot instans. Sebaiknya gunakan ID volume EBS dan titik pemasangan untuk identifikasi perangkat yang konsisten.
- Volume EBS NVMe memiliki ID volume EBS yang ditetapkan sebagai nomor seri dalam identifikasi perangkat. Gunakan perintah `lsblk -o +SERIAL` untuk mencantumkan nomor seri.
- Format nama perangkat NVMe dapat bervariasi tergantung pada apakah volume EBS dilampirkan selama atau setelah peluncuran instans. Nama perangkat NVMe untuk volume yang dilampirkan setelah peluncuran instans menyertakan prefiks `/dev/`, sedangkan nama perangkat NVMe untuk volume yang dilampirkan selama peluncuran instans tidak menyertakan prefiks `/dev/`.
  - Untuk Amazon Linux atau FreeBSD AMI, gunakan `sudo ebsnvme-id /dev/nvme0n1 -u` perintah untuk nama perangkat NVMe yang konsisten.

- Untuk distribusi lain, gunakan perintah `sudo nvme id-ctrl -V /dev/nvme0n1` untuk menentukan nama perangkat NVMe. Anda mungkin perlu menyertakan opsi `--vendor-specific` perintah.
- Saat perangkat diformat, UUID akan dihasilkan yang akan bertahan selama masa pakai sistem file. Label perangkat dapat ditetapkan pada saat yang sama. Untuk informasi lebih lanjut, lihat Menyediakan volume Amazon EBS untuk digunakan dan Boot dari volume yang salah.

AMI Amazon Linux

Dengan AMI Amazon Linux 2017.09.01 atau yang lebih baru (termasuk Amazon Linux 2), Anda dapat menjalankan perintah ebsnvme-id sebagai berikut untuk memetakan nama perangkat NVMe ke ID volume dan nama perangkat:

Contoh berikut menunjukkan perintah dan output untuk volume yang dilampirkan selama peluncuran instans. Harap diperhatikan bahwa nama perangkat NVMe tidak menyertakan prefiks `/dev/`.

```
[ec2-user ~]$ sudo /sbin/ebsnvme-id /dev/nvme0n1
Volume ID: vol-01324f611e2463981
sda
```

Contoh berikut menunjukkan perintah dan output untuk volume yang dilampirkan setelah peluncuran instans. Harap diperhatikan bahwa nama perangkat NVMe menyertakan prefiks `/dev/`.

```
[ec2-user ~]$ sudo /sbin/ebsnvme-id /dev/nvme1n1
Volume ID: vol-064784f1011136656
/dev/sdf
```

Amazon Linux juga membuat tautan simbolik dari nama perangkat di pemetaan perangkat blok (misalnya, `/dev/sdf`), ke nama perangkat NVMe.

AMI FreeBSD

Memulai dengan FreeBSD 12.2-RELEASE, Anda dapat menjalankan perintah ebsnvme-id seperti yang ditunjukkan di atas. Berikan nama perangkat NVMe (misalnya `nvme0`) atau perangkat disk (misalnya `nvd0` atau `nda0`). FreeBSD juga membuat tautan simbolis ke perangkat disk (misalnya, `/dev/aws/disk/ebs/volume_id`).

AMI Linux Lainnya

Dengan versi kernel 4.2 atau lebih baru, Anda dapat menjalankan perintah nvme id-ctrl sebagai berikut untuk memetakan perangkat NVMe ke ID volume. Pertama, instal paket baris perintah NVMe, `nvme-cli`, menggunakan alat manajemen paket untuk distribusi Linux Anda. Untuk petunjuk pengunduhan dan penginstalan untuk distribusi lainnya, lihat dokumentasi khusus untuk distribusi Anda.

Contoh berikut mendapatkan ID volume dan nama perangkat NVMe untuk volume yang terpasang saat peluncuran instans. Harap diperhatikan bahwa nama perangkat NVMe tidak menyertakan prefiks `/dev/`. Nama perangkat tersedia melalui ekstensi khusus vendor pengendali NVMe (bite 384:4095 dari identifikasi pengendali):

```
[ec2-user ~]$ sudo nvme id-ctrl -V /dev/nvme0n1
NVME Identify Controller:
vid     : 0x1d0f
ssvid   : 0x1d0f
sn      : vol01234567890abcdef
mn      : Amazon Elastic Block Store
...
0000: 2f 64 65 76 2f 73 64 6a 20 20 20 20 20 20 20 20 "sda..."
```

Contoh berikut mendapatkan ID volume dan nama perangkat NVMe untuk volume yang terpasang saat peluncuran instans. Harap diperhatikan bahwa nama perangkat NVMe menyertakan prefiks `/dev/`.

```
[ec2-user ~]$ sudo nvme id-ctrl -V /dev/nvme1n1
NVME Identify Controller:
vid     : 0x1d0f
ssvid   : 0x1d0f
sn      : volabcdef01234567890
mn      : Amazon Elastic Block Store
...
0000: 2f 64 65 76 2f 73 64 6a 20 20 20 20 20 20 20 20 "/dev/sdf..."
```

Perintah lsblk mencantumkan perangkat yang tersedia dan titik pemasangannya (jika ada). Ini membantu Anda menentukan nama perangkat yang tepat untuk digunakan. Dalam contoh ini, `/dev/nvme0n1p1` dipasang sebagai perangkat root dan `/dev/nvme1n1` dilampirkan tetapi tidak terpasang.

```
[ec2-user ~]$ lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
```

```
nvme1n1       259:3    0  100G  0 disk
nvme0n1       259:0    0    8G  0 disk
  nvme0n1p1   259:1    0    8G  0 part /
  nvme0n1p128 259:2    0    1M  0 part
```

Instans Windows

Anda dapat menjalankan perintah **`ebsnvme-id`** untuk memetakan nomor disk perangkat NVMe untuk ID volume EBS dan nama perangkat. Secara default, semua perangkat NVMe EBS dienumerasi. Anda dapat melewatkan nomor disk untuk informasi enumerasi perangkat tertentu. `ebsnvme-id`Alat ini termasuk dalam AMI Windows Server terbaru yang AWS disediakan yang terletak di`C:\ProgramData\Amazon\Tools`.

Dimulai dengan paket `1.5.0`, driver AWS NVMe versi terbaru `ebsnvme-id` alat diinstal oleh paket driver. Versi terbaru hanya tersedia dalam paket driver. Tautan unduhan mandiri untuk alat `ebsnvme-id` ini tidak akan lagi menerima pembaruan. Versi terakhir yang tersedia melalui tautan mandiri adalah `1.1.0`, yang dapat diunduh menggunakan tautan ebsnvme-id.zip dan mengekstrak konten ke instans Amazon EC2 Anda untuk mendapatkan akses ke `ebsnvme-id.exe`.

```
PS C:\ProgramData\Amazon\Tools> ebsnvme-id.exe
Disk Number: 0
Volume ID: vol-0d6d7ee9f6e471a7f
Device Name: sda1

Disk Number: 1
Volume ID: vol-03a26248ff39b57cf
Device Name: xvdd

Disk Number: 2
Volume ID: vol-038bd1c629aa125e6
Device Name: xvde

Disk Number: 3
Volume ID: vol-034f9d29ec0b64c89
Device Name: xvdb

Disk Number: 4
Volume ID: vol-03e2dbe464b66f0a1
Device Name: xvdc
```

```
PS C:\ProgramData\Amazon\Tools> ebsnvme-id.exe 4
```

```
Disk Number: 4
Volume ID: vol-03e2dbe464b66f0a1
Device Name: xvdc
```

Waktu tunggu I/O operasi NVMe untuk volume Amazon EBS

Sebagian besar sistem operasi menentukan batas waktu untuk I/O operasi yang dikirimkan ke perangkat NVMe.

Instans Linux

Di Linux, volume EBS yang dilampirkan ke Nitro-based instance menggunakan driver NVMe default yang disediakan oleh sistem operasi. Sebagian besar sistem operasi menentukan batas waktu untuk I/O operasi yang dikirimkan ke perangkat NVMe. Waktu habis adalah 30 detik dan dapat diubah menggunakan parameter boot `nvme_core.io_timeout`. Untuk sebagian besar kernel Linux sebelum versi 4.6, parameter ini adalah `nvme.io_timeout`.

Jika I/O latensi melebihi nilai parameter batas waktu ini, driver NVMe Linux gagal I/O dan mengembalikan kesalahan ke sistem file atau aplikasi. Tergantung pada I/O operasi, sistem file atau aplikasi Anda dapat mencoba lagi kesalahan. Dalam beberapa kasus, sistem file Anda mungkin dipasang ulang sebagai hanya-baca.

Untuk pengalaman yang serupa dengan volume EBS yang dilampirkan pada instans Xen, kami menyarankan agar mengatur `nvme_core.io_timeout` ke nilai tertinggi yang mungkin. Untuk kernel saat ini, maksimalnya adalah 4294967295, sedangkan untuk kernel sebelumnya maksimal adalah 255. Tergantung pada versi Linux, batas waktu mungkin sudah diatur ke nilai maksimum yang mendukung. Misalnya, batas waktu diatur ke 4294967295 secara default untuk AMI Amazon Linux 2017.09.01 dan yang lebih baru.

Anda dapat memverifikasi nilai maksimum untuk distribusi Linux Anda dengan menulis nilai yang lebih tinggi dari nilai maksimum hingga `/sys/module/nvme_core/parameters/io_timeout` yang disarankan dan memeriksa kesalahan Hasil numerik di luar rentang saat mencoba untuk menyimpan file.

Instans Windows

Pada Windows, batas waktu default adalah 60 detik dan maksimum adalah 255 detik. Anda dapat memodifikasi pengaturan registri kelas disk `TimeoutValue` menggunakan prosedur yang diuraikan dalam Entri Daftar untuk Driver SCSI Miniport.

NVMe Batalkan perintah untuk volume Amazon EBS

Per `Abort` intah ini adalah perintah admin NVMe yang dikeluarkan untuk mengakhiri perintah tertentu yang sebelumnya dikirimkan ke pengontrol. Perintah ini biasanya dikeluarkan oleh driver perangkat ke perangkat penyimpanan yang telah melebihi ambang batas waktu I/O operasi.

Jenis instans Amazon EC2 yang mendukung `Abort` perintah secara default akan mengakhiri perintah tertentu yang sebelumnya dikirimkan ke pengontrol saat `Abort` perintah dikeluarkan untuk volume Amazon EBS terlampir. Instans Amazon EC2 yang tidak mendukung `Abort` perintah tidak mengambil tindakan saat `Abort` perintah dikeluarkan untuk volume Amazon EBS terlampir.

Per `Abort` intah ini didukung dengan:

- Perangkat Amazon EBS dengan perangkat NVMe versi 1.4 atau lebih tinggi.
- Semua instans Amazon EC2, kecuali jenis Xen-based instans dan jenis Nitro-based instans berikut:
  - Tujuan umum: A1 | M5 | M5a | M5ad | M5d | M5dn | M5n | M5Zn | M6g | M6gd | Mac1 | Mac2 | T3 | T3a | T4g
  - Komputasi dioptimalkan: C5 | c5a | C5ad | C5d | C5n | C6g | C6gd
  - Memori dioptimalkan: R5 | R5a | R5ad | R5d | R5dn | R5n | R6g | R6gd | | | | | | X2Gd | X2ieZn | Z1d U-12tb1 U-18tb1 U-24tb1 U-3tb1 U-6tb1 U-9tb1
  - Penyimpanan dioptimalkan: D3 | D3en | i3en
  - Komputasi yang dipercepat: DL1 | G4ad | G4dn | G5 | G5g | Inf1 | P3dN | P4d | P4de | VT1

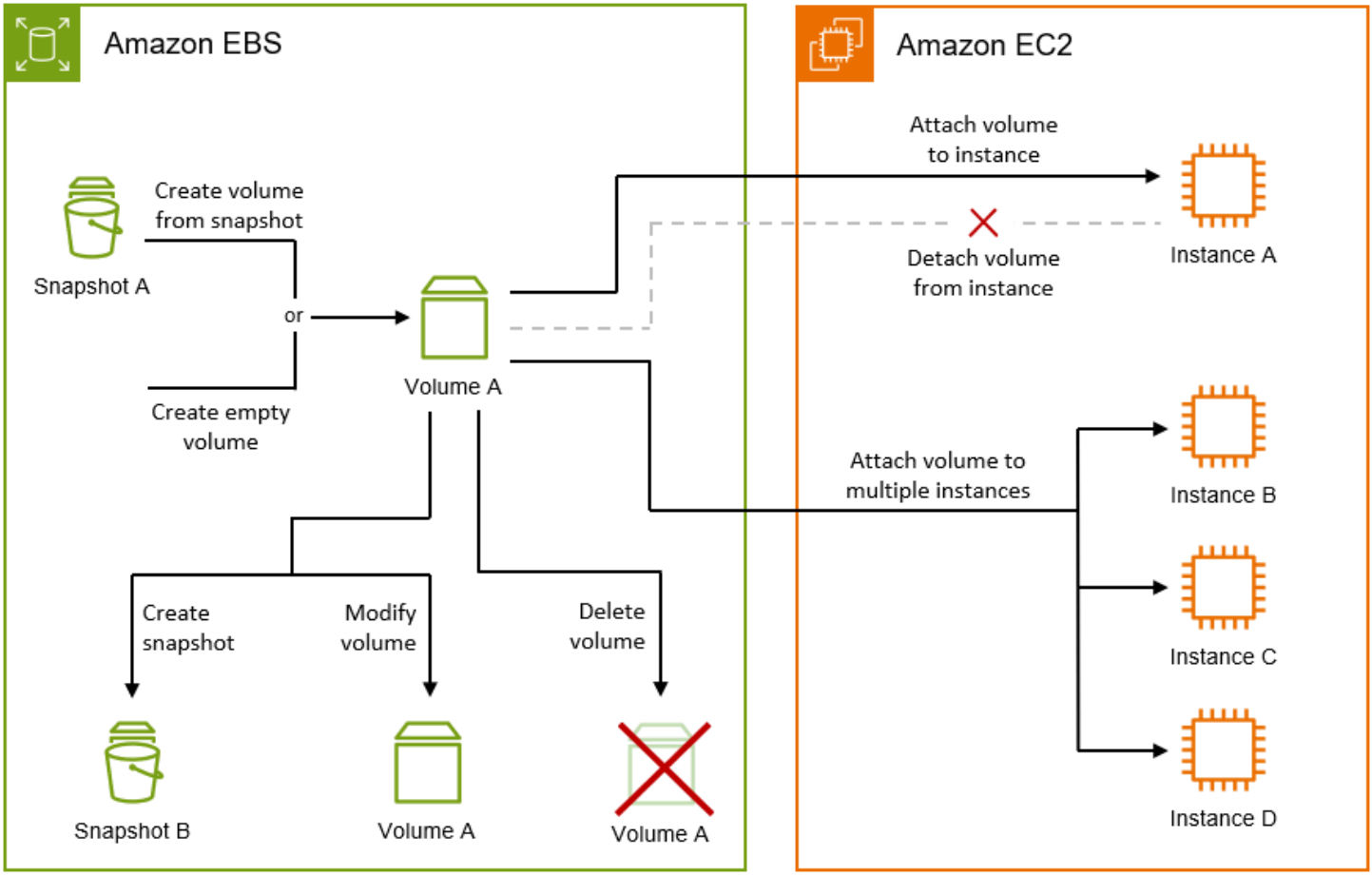
Untuk informasi lebih lanjut, lihat Abort perintah bagian 5.1 dari Spesifikasi Das ar NVM Express.

Siklus hidup volume Amazon EBS

Siklus hidup volume Amazon EBS dimulai dengan proses pembuatan. Anda dapat membuat volume dari snapshot Amazon EBS atau Anda dapat membuat volume kosong. Sebelum Anda dapat menggunakan volume, Anda harus melampirkannya ke satu atau beberapa instans Amazon EC2 yang berada di Zona Ketersediaan yang sama dengan volume. Anda dapat melampirkan beberapa volume ke sebuah instance. Jika diperlukan, Anda dapat melepaskan volume dari satu instance dan kemudian melampirkannya ke instance lain. Jika persyaratan penyimpanan Anda berubah, Anda dapat mengubah ukuran atau kinerja volume kapan saja. Anda dapat membuat cadangan volume

secara point-in-time dengan membuat snapshot Amazon EBS. Jika Anda tidak lagi membutuhkan volume, Anda dapat menghapusnya untuk berhenti menimbulkan biaya penyimpanan terkait.

Gambar berikut menunjukkan tindakan yang dapat Anda lakukan pada volume Anda sebagai bagian dari siklus hidup volume. Ada juga tugas yang Anda lakukan dengan menghubungkan ke instance dan menjalankan perintah sistem operasi. Misalnya, memformat volume, memasang volume, mengelola partisi, dan melihat ruang disk kosong.



Tugas

- Buat volume Amazon EBS
- Bagikan volume
- Menyalin volume Amazon EBS
- Melampirkan volume Amazon EBS ke instans Amazon EC2
- Melampirkan volume EBS ke beberapa instans EC2 menggunakan Multi-Attach
- Menyediakan volume Amazon EBS untuk digunakan
- Melihat informasi tentang volume Amazon EBS

- Memodifikasi volume Amazon EBS menggunakan operasi Volume Elastic
- Melepaskan volume Amazon EBS dari instans Amazon EC2
- Menghapus volume Amazon EBS

Buat volume Amazon EBS

Anda dapat membuat sebuah volume Amazon EBS lalu melampirkannya ke instans EC2 apa pun di Zona Ketersediaan yang sama.

Anda dapat membuat volume kosong, atau Anda dapat membuat volume dari snapshot Amazon EBS. Jika Anda membuat volume dari snapshot, volume dimulai sebagai replika yang tepat dari volume yang digunakan untuk membuat snapshot tersebut.

Inisialisasi volume

Saat Anda membuat volume dari snapshot, blok penyimpanan dari snapshot harus diunduh dari Amazon S3 dan ditulis ke volume sebelum Anda dapat mengaksesnya. Proses ini disebut inisialisasi volume. Selama waktu ini, volume akan mengalami peningkatan I/O latensi. Kinerja volume penuh dicapai hanya setelah semua blok penyimpanan diunduh dan ditulis ke volume.

Tingkat inisialisasi volume default berfluktuasi sepanjang proses inisialisasi, yang dapat membuat waktu penyelesaian tidak dapat diprediksi.

Untuk meminimalkan dampak kinerja yang terkait dengan inisialisasi volume, Anda dapat menggunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume (tingkat inisialisasi volume) atau pemulihan snapshot cepat. Untuk informasi selengkapnya, lihat Inisialisasi volume Amazon EBS.

Enkripsi volume

Status enkripsi volume tergantung pada apakah akun Anda di aktifkan untuk enkripsi secara default, dan pada status enkripsi snapshot, jika Anda memilih untuk menggunakannya. Tabel berikut merangkum kemungkinan hasil enkripsi.

Enkripsi secara default	Snapshot digunakan?	Hasil enkripsi volume	Catatan
Nonaktif	Tidak	Enkripsi opsional	Jika Anda mengaktifkan enkripsi, Anda dapat menentukan kunci KMS yang akan digunakan. Jika Anda mengaktifkan enkripsi tetapi tidak menentukan kunci KMS, Kunci yang dikelola AWS (`aws/ebs`) digunakan.
Nonaktif	Ya, tidak terenkripsi	Enkripsi opsional	Jika Anda mengaktifkan enkripsi, Anda dapat menentukan kunci KMS yang akan digunakan. Jika Anda mengaktifkan enkripsi tetapi tidak menentukan kunci KMS, Kunci yang dikelola AWS (`aws/ebs`) digunakan.
Nonaktif	Ya, terenkripsi	Enkripsi otomatis	Anda dapat menentukan kunci KMS untuk digunakan. Jika Anda tidak menentukan kunci KMS, volume dienkripsi menggunakan kunci KMS yang sama dengan snapshot sumber.
Diaktifkan	Tidak	Enkripsi otomatis	Anda dapat menentukan kunci KMS untuk digunakan. Jika Anda tidak menentukan kunci KMS, kunci yang ditentukan untuk enkripsi secara default digunakan.
Diaktifkan	Ya, tidak terenkripsi	Enkripsi otomatis	Anda dapat menentukan kunci KMS untuk digunakan. Jika Anda tidak menentukan kunci KMS, kunci yang ditentukan untuk enkripsi secara default digunakan.
Diaktifkan	Ya, terenkripsi	Enkripsi otomatis	Anda dapat menentukan kunci KMS untuk digunakan. Jika Anda tidak menentukan kunci KMS, volume dienkripsi menggunakan kunci yang sama dengan snapshot

Enkripsi secara default	Snapshot digunakan?	Hasil enkripsi volume	Catatan
			sumber (konsol), atau kunci yang ditentukan untuk enkripsi secara default (CLI/API).

Pertimbangan tambahan

- Volume harus dilampirkan ke instans di Zona Ketersediaan yang sama.
- Volume siap digunakan hanya setelah memasuki `available` negara bagian.
- Saat Anda membuat volume menggunakan konsol, `gp3` adalah jenis volume default. Untuk alat baris perintah, API, dan SDK, `gp2` adalah tipe volume default.
- Untuk menggunakan volume dengan instance yang berjalan di pos terdepan, Anda harus membuat volume di pos terdepan yang sama dengan instance.
- Jika Anda membuat volume untuk digunakan dengan instance Windows, dan lebih besar dari 2048 GiB, pastikan Anda mengonfigurasi volume untuk menggunakan tabel partisi GPT. Untuk informasi selengkapnya, lihat Batasan volume Amazon EBS dan dukungan Windows untuk disk yang lebih besar dari 2 TB. .
- Volume juga dibuat secara tidak langsung dengan meluncurkan instance Amazon EC2. Baik AMI yang digunakan untuk meluncurkan instance, atau permintaan peluncuran instans itu sendiri dapat menyertakan pemetaan perangkat blok untuk volume Amazon EBS. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat.

Console

Untuk membuat volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume lalu pilih Buat volume.
3. (Outpostkhusus pelanggan) Outpost Untuk ARN, masukkan ARN tempat untuk membuat volume. AWS Outpost
4. Untuk Tipe volume, pilih tipe volume yang akan dibuat. Untuk informasi selengkapnya tentang jenis volume yang tersedia, lihatTipe volume Amazon EBS.

5. Untuk Ukuran, masukkan ukuran volume, dalam satuan GiB. Untuk informasi selengkapnya, lihat Batasan volume Amazon EBS.
6. (Untuk*io1*,*io2*, dan *gp3* hanya) Untuk IOPS, masukkan jumlah maksimum input/output operasi per detik (IOPS) yang harus disediakan volume.
7. (*gp3*Hanya untuk) Untuk Th roughput, masukkan throughput yang harus disediakan volume, di MiB/s.
8. Untuk Zona Ketersediaan, pilih Zona Ketersediaan tempat pembuatan volume.
9. Untuk ID Snapshot, lakukan salah satu hal berikut:
   - Untuk membuat volume kosong, pertahankan nilai default (Jangan membuat volume dari snapshot).
   - Untuk membuat volume dari snapshot, pilih snapshot yang akan digunakan.
10. Jika Anda telah memilih snapshot, untuk Tingkat inisialisasi Volume, Anda dapat secara opsional menentukan tingkat inisialisasi volume MiB/s, di, di mana blok snapshot akan diunduh dari Amazon S3 ke volume setelah pembuatan. Untuk informasi selengkapnya, lihat Gunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume. Untuk menggunakan tingkat inisialisasi default atau pemulihan snapshot cepat (jika diaktifkan untuk snapshot yang dipilih), jangan tentukan tingkat.
11. (*io1*dan *io2* hanya) Untuk mengaktifkan volume untuk Amazon EBS Multi-Attach, pilih Akti Multi-Attach fkan. Untuk informasi selengkapnya, lihat Melampirkan volume EBS ke beberapa instans EC2 menggunakan Multi-Attach.
12. Atur status enkripsi untuk volume.
    - Jika akun Anda diaktifkan untuk enkripsi secara default, enkripsi otomatis dan tidak dapat dinonaktifkan.
    - Jika Anda memilih snapshot terenkripsi, enkripsi otomatis dan tidak dapat dinonaktifkan.
    - Jika akun Anda tidak diaktifkan untuk enkripsi secara default, dan Anda memilih snapshot yang tidak dienkripsi atau tidak memilih snapshot, enkripsi bersifat opsional.
13. (Opsional) Untuk menetapkan tag khusus ke volume, di bagian Tag, pilih Tambah tag, lalu masukkan kunci tag dan pasangan nilai.
14. Pilih Buat volume.
15. Untuk menggunakan volume, tunggu sampai mencapai status, lalu `available` lampirkan ke instans Amazon EC2 di Zona Ketersediaan yang sama. Untuk informasi selengkapnya, lihat Melampirkan volume Amazon EBS ke instans Amazon EC2.

AWS CLI

Untuk membuat volume

Gunakan perintah create-volume. Contoh berikut membuat volume gp3 kosong dengan ukuran 100 GiB di Zona Ketersediaan yang ditentukan.

```
aws ec2 create-volume \
    --volume-type gp3 \
    --size 100 \
    --availability-zone us-east-1a
```

PowerShell

Untuk membuat volume

Gunakan New-EC2Volume cmdlet. Contoh berikut membuat volume gp3 kosong dengan ukuran 100 GiB di Zona Ketersediaan yang ditentukan.

```
New-EC2Volume `
    -VolumeType gp3 `
    -Size 100 `
    -AvailabilityZone us-east-1a
```

Bagikan volume

Anda dapat menggunakan AWS Resource Access Manager (AWS RAM) untuk berbagi volume Amazon EBS dengan AWS akun lain. Setelah Anda berbagi volume, akun pengguna dapat melihat informasi volume dan membuat salinan volume bersama di Zona Ketersediaan yang sama. Untuk informasi selengkapnya tentang membuat salinan, lihatMenyalin volume Amazon EBS. Akun yang dikonsumsi tidak dapat melampirkan, memodifikasi, menghapus, atau membuat snapshot dari volume bersama.

Misalnya, Anda dapat berbagi volume produksi dengan akun pengembangan untuk menyegarkan lingkungan pengujian dengan data produksi terbaru sambil mempertahankan isolasi tingkat akun.

Anda dapat membagikan volume yang:

- Tidak terenkripsi
- Dienkripsi dengan kunci yang dikelola pelanggan

Anda tidak dapat membagikan volume yang dienkripsi dengan kunci terkel AWS ola default untuk Amazon EBS.

Daftar Isi

- Izin
- Bagikan volume
- Lihat volume bersama
- Pertimbangan untuk berbagi volume
- Memantau aktivitas berbagi
- Volume bersama dan Recycle Bin
- Harga

Izin

Saat Anda membuat berbagi sumber daya untuk volume bersama, Anda harus mengaitkan izin terkelola dengan berbagi sumber daya. Izin terkelola menentukan tindakan yang dapat dilakukan akun yang menggunakan akun pada volume bersama dalam pembagian sumber daya. AWS RAM menyediakan izin terkel AWS ola berikut:

- AWSRAMDefaultPermissionEBSVolume(default) — Memberikan izin akun pengguna untuk melihat informasi volume.
- AWSRAMPermissionEBSVolumeCopyAccess— Memberikan izin akun konsumen untuk melihat informasi volume dan membuat salinan volume bersama.

Atau, Anda dapat membuat izin yang dikelola pelanggan untuk jenis `ec2:Volume` sumber daya. Untuk informasi selengkapnya, lihat Membuat dan menggunakan izin yang dikelola pelanggan di Panduan AWS RAM Pengguna.

Bagikan volume

Gunakan salah satu metode berikut untuk berbagi volume Amazon EBS.

Console

Untuk berbagi volume

1. Buka konsol Amazon Elastic Compute Cloud dihttps://console.aws.amazon.com/ec2/.

2. Pada panel navigasi, pilih Volume.
3. Pilih volume, lalu pilih Tindakan , Bagikan volume.
4. Lakukan salah satu tindakan berikut:
   - Untuk menambahkan volume ke berbagi sumber daya yang ada, pilih berbagi sumber daya.
   - Untuk membuat berbagi sumber daya baru, pilih Buat berbagi sumber daya baru. Ini membuka AWS RAM konsol, tempat Anda mengonfigurasi berbagi.
5. Untuk Kepala Sekolah, tambahkan akun untuk berbagi volume. Anda dapat menambahkan ID akun individu, unit organisasi (OU), atau seluruh organisasi Anda.
6. Untuk Iz in ter kelola, pilih salah satu dari berikut ini:
   - AWSRAMDefaultPermissionEBSVolume(default) — Akun yang dikonsumsi hanya dapat melihat metadata volume.
   - AWSRAMPermissionEBSVolumeCopyAccess— Akun yang dikonsumsi dapat melihat metadata volume dan membuat salinan volume bersama.
   - Izin yang dikelola pelanggan yang Anda buat untuk jenis `ec2:Volume` sumber daya.
7. Pilih Buat berbagi sumber daya.

AWS CLI

Gunakan perintah create-resource-share. Tentukan volume ARN, akun konsumsi, dan izin ARN. Untuk menemukan volume ARN, panggil describe- volume. Tanggapan mencakup `VolumeArn` bidang untuk setiap volume.

Contoh berikut membuat pembagian sumber daya yang berbagi volume `vol-1234567890abcdef0` dengan akun`222222222222`, dan menetapkan izin terkel `AWSRAMPermissionEBSVolumeCopyAccess` ola:

```
aws ram create-resource-share \
--name my-volume-share \
--resource-arns arn:aws:ec2:us-east-1:111111111111:volume/vol-1234567890abcdef0 \
--principals 222222222222 \
--permission-arns arn:aws:ram::aws:permission/AWSRAMPermissionEBSVolumeCopyAccess \
--region us-east-1
```

Untuk berbagi dengan izin default (`AWSRAMDefaultPermissionEBSVolume`, jelaskan saja), hilangkan `--permission-arns` parameter:

```
aws ram create-resource-share \
--name my-volume-share \
--resource-arns arn:aws:ec2:us-east-1:111111111111:volume/vol-1234567890abcdef0 \
--principals 222222222222 \
--region us-east-1
```

Untuk informasi selengkapnya tentang pembagian sumber daya, lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

Lihat volume bersama

Setelah Anda membuat berbagi sumber daya, akun yang ditentukan menerima undangan. Akun di luar organisasi Anda AWS Organizations harus menerima undangan sebelum dapat mengakses volume bersama. Untuk informasi selengkapnya, lihat Menerima dan menolak undangan berbagi sumber daya di Panduan AWS RAM Pengguna.

Anda dapat melihat volume yang dibagikan dengan akun Anda menggunakan salah satu metode berikut.

Console

Untuk melihat volume bersama

1. Buka konsol Amazon Elastic Compute Cloud dihttps://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.
3. Lakukan salah satu tindakan berikut:
   - Pilih filter preset Di bagikan dengan saya untuk menampilkan semua volume yang dibagikan dengan akun Anda.
   - Pilih Tambah filter, ID Pemilik, lalu masukkan ID akun berbagi.

Volume bersama menampilkan ID akun berbagi di kolom ID Pemilik.

AWS CLI

Gunakan perintah describe-volume dengan `owner-id` filter untuk menampilkan volume yang dibagikan dengan Anda oleh akun tertentu.

```
aws ec2 describe-volumes \
--filters Name=owner-id,Values=111111111111 \
--region us-east-1
```

Volume bersama menyer `OwnerId` takan bidang dalam respons, yang menunjukkan ID akun berbagi.

Pertimbangan untuk berbagi volume

Pertimbangkan hal berikut saat Anda berbagi volume dengan akun lain.

- Anda dapat berbagi volume dan volume yang tidak dienkripsi yang dienkripsi dengan kunci yang dikelola pelanggan. Anda tidak dapat membagikan volume yang dienkripsi dengan kunci ter AWS kelola default untuk Amazon EBS.
- Untuk berbagi volume, akun pemilik volume harus memiliki `kms:DescribeKey` izin pada kunci enkripsi Amazon EBS default. Ini diperlukan untuk semua volume, termasuk volume yang tidak dienkripsi, karena layanan memvalidasi status enkripsi volume selama proses berbagi.
- Untuk berbagi volume yang dienkripsi dengan kunci yang dikelola pelanggan, Anda juga harus membagikan kunci KMS dengan akun pengguna. Untuk informasi selengkapnya, lihat Mengizinkan pengguna di akun lain untuk menggunakan kunci KMS di Panduan Developer AWS Key Management Service .
- Untuk membuat salinan volume terenkripsi bersama, akun pengguna memerlukan`kms:CreateGrant`,,`kms:GenerateDataKey`, `kms:GenerateDataKeyWithoutPlaintext``kms:ReEncrypt*`, dan `kms:Decrypt` pada kunci KMS.
- Akun yang dikonsumsi tidak dapat melampirkan, memodifikasi, menghapus, atau membuat snapshot dari volume bersama. Mereka hanya dapat melihat metadata volume dan membuat salinan.
- Hanya satu operasi penyalinan yang dapat berlangsung pada volume bersama pada satu waktu, di semua akun yang digunakan untuk berbagi volume. Permintaan salinan tambahan mengembalikan kesalahan hingga operasi saat ini selesai.

- Nama Zona Ketersediaan (sepertius-east-1a) dipetakan ke lokasi fisik yang berbeda di AWS akun yang berbeda. Gunakan ID Zona Ketersediaan (sepertiuse1-az1) untuk mengidentifikasi lokasi fisik yang sama di seluruh akun. Untuk informasi selengkapnya, lihat ID Zona Keter sediaan untuk AWS sumber daya Anda di Panduan AWS RAM Pengguna.
- Menghapus volume dari pembagian sumber daya tidak membatalkan operasi penyalinan yang sedang berlangsung.
- Menghapus volume sumber tidak membatalkan operasi penyalinan yang sedang berlangsung.

Memantau aktivitas berbagi

AWS CloudTrail Pencatatan API

Semua panggilan API berbagi volume dan salin masuk AWS CloudTrail. CopyVolumes Panggilan API dicatat untuk akun yang memulai salinan dan akun pemilik volume. Untuk informasi selengkapnya, lihat Mencatat panggilan Amazon Elastic Compute Cloud API dengan AWS CloudTrail.

Pember EventBridge itahuan Amazon

Amazon EBS mengirimkan EventBridge acara Amazon saat operasi penyalinan lintas akun selesai. Baik pemilik volume maupun akun konsumen menerima acara tersebut. Peristiwa penyalinan volume menggunakan tipe detail EBS Volume Notification dengan nama acarasharedVolumeCopy. Acara ini mencakup ID volume, status penyelesaian (completedataufailed), dan stempel waktu.

Untuk mengatur notifikasi, buat EventBridge aturan Amazon yang mencocok EBS Volume Notification kan acara dengan nama sharedVolumeCopy acara. Untuk informasi selengkapnya, lihat EventBridge acara Amazon untuk Amazon EBS.

Volume bersama dan Recycle Bin

Jika Anda memiliki aturan retensi Recycle Bin yang berlaku untuk volume, menghapus volume bersama memindahkannya ke Recycle Bin. Volume tetap dalam pembagian AWS RAM sumber dayanya selama periode retensi. Saat Anda mengembalikan volume dari Recycle Bin, volume tersebut dapat diakses melalui pembagian sumber daya yang sama sebelum dihapus.

Perhatikan perilaku berbagi sumber daya berikut untuk volume yang ada di Recycle Bin:

- Jika Anda menghapus akun dari pembagian sumber daya saat volume berada di Recycle Bin, memulihkan volume tidak akan memberikan kembali akses akun tersebut.

- Jika Anda menambahkan akun ke berbagi sumber daya saat volume berada di Recycle Bin, akun tersebut memiliki akses ke volume setelah pemulihan.

Setelah memulihkan volume dari Recycle Bin, verifikasi bahwa hanya akun yang dituju yang memiliki akses dengan meninjau pembagian sumber daya Anda di AWS RAM konsol.

Harga

Tidak ada biaya untuk berbagi volume melalui AWS RAM. Akun yang membuat salinan membayar biaya salinan volume dan biaya volume Amazon EBS reguler untuk volume baru. Untuk informasi selengkapnya, lihat harga Amazon EBS.

Menyalin volume Amazon EBS

Salinan volume adalah salinan volume sumber yang konsisten terhadap crash, point-in-time. Ini mencakup semua blok data yang ditulis ke volume sumber pada saat inisialisasi salinan volume dimulai. Salinan volume mendapatkan ID volume uniknya sendiri. Salinan volume dibuat segera dan dapat dilampirkan ke instans Amazon EC2 setelah mencapai `available` status. Dengan menggunakan volume copy, Anda dapat dengan cepat menyalin data produksi untuk lingkungan pengujian dan pengembangan.

Anda juga dapat membuat salinan dari volume yang dibagikan dengan Anda oleh AWS akun lain. Untuk menyalin volume dari akun lain, volume sumber harus dibagikan dengan Anda melalui AWS Resource Access Manager (AWS RAM). Untuk informasi selengkapnya, lihat Bagikan volume.

Inisialisasi

Salinan volume diinisialisasi setelah pembuatan. Selama inisialisasi, blok data disalin dari volume sumber dan ditulis ke salinan volume di latar belakang. Volume tetap dalam `initializing` status sampai inisialisasi selesai.

Kinerja selama inisialisasi

Operasi penyalinan tidak mempengaruhi kinerja volume sumber. Anda dapat terus menggunakan volume sumber secara normal selama proses penyalinan. Volume yang disalin dapat diakses secara instan tanpa menunggu data disalin dari volume sumber. Salinan volume menyediakan akses instan ke data dengan latensi milidetik satu digit. Namun, latensi aktual mungkin bervariasi tergantung pada jenis volume. Selama inisialisasi, salinan volume memberikan kinerja dasar yang sama dengan yang terendah dari tiga nilai berikut:

- 3.000 input/output operasi per detik (IOPS) dan 125 MiB/s
- Kinerja yang disediakan untuk volume sumber
- Performa yang disediakan untuk salinan volume

Salinan volume dapat melebihi kinerja dasar jika kriteria berikut terpenuhi:

1. Volume sumber dan salinan volume disediakan dengan lebih dari 3.000 IOPS dan 125. MiB/s
2. Volume sumber memiliki kapasitas kinerja yang tidak dimanfaatkan (kinerja yang didorong kurang dari kinerja yang disediakan).

Misalnya, jika volume sumber disediakan dengan 10.000 IOPS dan beban kerja Anda saat ini hanya mendorong 5.000 IOPS, dan salinan volume disediakan dengan 10.000 IOPS, salinan volume dapat mencapai kinerja yang lebih tinggi dari kinerja dasar 3.000 IOPS selama inisialisasi dengan menggunakan 5.000 IOPS volume sumber yang tidak dimanfaatkan.

Durasi inisialisasi

Waktu yang dibutuhkan untuk menginisialisasi salinan volume tergantung pada ukuran data blok yang ditulis ke volume sumber pada saat membuat salinan volume. Salinan volume diinisialisasi berdasarkan upaya terbaik, dengan pedoman umum berikut. Untuk blok data 1 TiB pertama, inisialisasi volume membutuhkan waktu hingga 6 jam. Untuk setiap 1 TiB blok data berikutnya hingga 16 TiB, inisialisasi membutuhkan 1,2 jam per TiB. Untuk data tertulis yang lebih besar dari 16 TiB, inisialisasi membutuhkan waktu 24 jam.

Pantau kemajuan inisialisasi

Anda dapat memantau kemajuan inisialisasi menggunakan perintah describe-volume- AWS CLI status atau Amazon. EventBridge Untuk informasi selengkapnya, lihat Memantau status inisialisasi volume Amazon EBS.

Enkripsi

Saat Anda menyalin volume, enkripsi salinan tergantung pada status enkripsi volume sumber, enkripsi akun Anda secara default, dan `KmsKeyId` parameter `Encrypted` dan yang Anda tentukan dalam permintaan salin.

Volume sumber yang tidak dienkripsi

Jika volume sumber tidak dienkripsi dan enkripsi secara default dinonaktifkan untuk akun Anda, salinannya tidak dienkripsi. Anda dapat mengenkripsi salinan dengan mengatur `Encrypted` parameter `ketrue`. Jika Anda tidak menentukan a KMS key, kunci enkripsi Amazon EBS default akun Anda digunakan.

Jika volume sumber tidak dienkripsi dan enkripsi secara default diaktifkan untuk akun Anda, salinan akan dienkripsi secara otomatis. Jika Anda tidak menentukan a KMS key, kunci enkripsi Amazon EBS default akun Anda digunakan.

Volume sumber terenkripsi

Jika volume sumber dienkripsi, salinan selalu dienkripsi. Anda tidak dapat membuat salinan yang tidak dienkripsi dari volume sumber terenkripsi.

Untuk salinan akun yang sama, salinan menggunakan kunci KMS yang sama dengan volume sumber secara default. Anda dapat menentukan kunci yang dikelola pelanggan yang berbeda untuk mengenkripsi ulang salinan.

Untuk salinan lintas akun, salinan menggunakan kunci enkripsi Amazon EBS default akun Anda secara default. Anda dapat menentukan kunci yang dikelola pelanggan yang berbeda dalam `KmsKeyId` parameter. Anda dapat menggunakan kunci KMS akun sumber untuk mengenkripsi salinan volume jika kunci tersebut telah dibagikan dengan akun Anda.

Cross-account persyaratan enkripsi

Untuk menyalin volume bersama yang dienkripsi dengan kunci yang dikelola pelanggan, pemilik volume harus membagikan kunci KMS dengan akun Anda. Untuk informasi selengkapnya, lihat Mengiz inkan pengguna di akun lain menggunakan a KMS key di Panduan AWS Key Management Service Pengembang.

Volume yang dienkripsi dengan kunci AWS terkelola default untuk Amazon EBS tidak dapat dibagikan atau disalin di seluruh akun.

Hasil enkripsi

Volume sumber yang tidak dienkripsi

Enkripsi secara default	Anda menentukan a KMS key	Hasil
Nonaktif	Tidak	Salinan tidak dienkripsi

Enkripsi secara default	Anda menentukan a KMS key	Hasil
Nonaktif	Ya	Salin dienkripsi dengan kunci yang ditentukan
Diaktifkan	Tidak	Salin terenkripsi dengan kunci enkripsi Amazon EBS default akun Anda
Diaktifkan	Ya	Salin dienkripsi dengan kunci yang ditentukan

Volume sumber terenkripsi (akun yang sama)

Anda menentukan a KMS key	Hasil
Tidak	Salin dienkripsi dengan kunci yang sama dengan volume sumber
Ya	Salin dienkripsi ulang dengan kunci yang ditentukan

Volume sumber terenkripsi (lintas akun)

Anda menentukan a KMS key	Hasil
Tidak	Salin terenkripsi dengan kunci enkripsi Amazon EBS default akun Anda
Ya	Salin dienkripsi dengan kunci yang ditentukan

Cross-account salinan tidak pernah menggunakan kembali kunci KMS akun sumber secara default. Jika Anda ingin menggunakan kunci dari akun sumber, kunci tersebut harus dibagikan dengan akun Anda.

Pertimbangan-pertimbangan

- Salinan volume menangkap status point-in-time volume sumber yang konsisten dengan crash. Ini hanya mencakup data yang telah ditulis ke volume pada saat permintaan salinan. Itu tidak termasuk data yang di-cache oleh aplikasi atau sistem operasi. Untuk mendapatkan salinan yang konsisten dengan aplikasi, jeda penulisan ke volume atau matikan aplikasi sebelum membuat salinan. Untuk database, gunakan mekanisme yang sesuai untuk membersihkan halaman kotor dan membeku I/O kan (seperti `fsfreeze` pada Linux atau VSS di Windows). Anda dapat melanjutkan penulisan setelah permintaan salinan kembali.
- Anda hanya dapat membuat satu salinan volume dari volume sumber pada satu waktu. Anda dapat membuat salinan berikutnya dari volume sumber yang sama hanya setelah salinan volume sebelumnya telah sepenuhnya diinisialisasi.
- Anda dapat memiliki maksimal 5 salinan volume dalam proses per Wilayah. Jika Anda melebihi kuota ini, Anda mendapatkan `CopyVolumesLimitExceeded` kesalahan. Anda dapat meminta kenaikan kuota jika diperlukan.
- Salinan volume harus dibuat di Zona Ketersediaan yang sama dengan volume sumber.
- Ukuran salinan volume harus sama dengan atau lebih besar dari ukuran volume sumber.
- Anda tidak dapat menyalin salinan volume saat sedang dibuat atau diinisialisasi.
- Untuk membuat salinan volume, volume sumber harus dalam `in-use` status `available` atau, dan modifikasi volume harus dalam `optimizing` status `completed` atau.
- Salinan volume tunduk pada akun dan penyimpanan regional dan kuota IOPS yang sama dengan volume Amazon EBS biasa. Untuk informasi selengkapnya, lihat kuota Amazon EBS.
- Jika Anda menghapus volume sumber saat operasi penyalinan sedang berlangsung, operasi penyalinan masih selesai.
- Tag yang ditetapkan ke volume sumber tidak ditetapkan ke salinan volume.
- Anda tidak dapat membuat salinan dari volume di Outpost atau di Zona Panjang Gelombang.

Harga

Saat Anda membuat salinan volume sumber, Anda akan dikenakan biaya untuk operasi penyalinan. Biaya operasi penyalinan didasarkan pada ukuran blok data yang ditulis ke volume sumber pada saat pembuatan salinan. Volume yang baru dibuat dikenakan biaya dengan cara yang sama seperti volume EBS Amazon lainnya. Untuk informasi selengkapnya, lihat harga Amazon EBS.

Menyalin volume

Gunakan salah satu metode berikut untuk menyalin volume Amazon EBS.

Console

Untuk menyalin volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.
3. Pilih volume yang akan disalin dan pilih Tind akan, Salin volume.
4. Untuk Jenis Volume, pilih jenis volume untuk salinan. Jenis volume default adalah gp3.
5. Untuk Ukuran, masukkan ukuran untuk salinan volume, di GiBs. Ukuran harus sama dengan atau lebih besar dari ukuran volume sumber.
6. (*io1*,*io2*, dan *gp3* hanya) Untuk IOPS, masukkan jumlah maksimum input/output operasi per detik (IOPS) untuk salinan volume.
7. (*gp3*hanya) Untuk Th roughput, masukkan throughput untuk salinan volume, di MiB/s.
8. (*io1*dan *io2* hanya) Untuk mengaktifkan salinan volume untuk Amazon EBS Multi-Attach, pilih Akti Multi-Attach fkan.
9. Untuk Enkripsi, default tergantung pada volume sumber dan pengaturan akun Anda. Untuk salinan akun yang sama, salinan mewarisi kunci enkripsi volume sumber. Untuk salinan lintas akun, kunci enkripsi Amazon EBS default akun Anda digunakan. Anda dapat memilih kunci yang dikelola pelanggan yang berbeda.
10. (Opsional) Untuk menetapkan tag khusus ke salinan volume, di bagian Tag, pilih Tambah tag, lalu masukkan kunci tag dan pasangan nilai.
11. Pilih Salin volume.
12. Volume yang disalin memasuki `creating` status dan kemudian beralih ke `available` segera setelahnya. Anda kemudian dapat melampirkannya ke instans Amazon EC2 di Zona Ketersediaan yang sama.

AWS CLI

Untuk menyalin volume

Gunakan perintah copy-volume.

Contoh berikut membuat salinan volume `vol-01234567890abcdef` dengan jenis `gp3` volume, ukuran `100` GiB, dan throughput dari `250` MiB/s.

```
aws ec2 copy-volumes \
--source-volume-id vol-01234567890abcdef \
--volume-type gp3 \
--size 100 \
--throughput 250
```

Untuk mengenkripsi volume yang tidak dienkripsi saat menyalin

Gunakan perintah copy-volume dengan parameter `--encrypted` dan`--kms-key-id`.

```
aws ec2 copy-volumes \
--source-volume-id vol-01234567890abcdef \
--encrypted \
--kms-key-id alias/my-key
```

Untuk mengenkripsi ulang dengan kunci yang dikelola pelanggan yang berbeda

Gunakan perintah copy-volume dengan `--kms-key-id` parameter untuk menentukan kunci yang berbeda.

```
aws ec2 copy-volumes \
--source-volume-id vol-01234567890abcdef \
--kms-key-id arn:aws:kms:us-east-1:222222222222:key/abcd1234-a123-456a-a12b-
a123b4cd56ef
```

PowerShell

Untuk menyalin volume

Gunakan Copy-EC2Volume cmdlet.

Contoh berikut membuat salinan volume `vol-01234567890abcdef` dengan jenis `gp3` volume, ukuran `100` GiB, dan throughput dari `250` MiB/s.

```
Copy-EC2Volume `
-SourceVolumeId vol-01234567890abcdef `
-VolumeType gp3 `
-Size 100 `
-Throughput 250
```

Untuk mengenkripsi volume yang tidak dienkripsi saat menyalin

Gunakan Copy-EC2Volume cmdlet dengan parameter `-Encrypted` dan`-KmsKeyId`.

```
Copy-EC2Volume `
-SourceVolumeId vol-01234567890abcdef `
-Encrypted $true `
-KmsKeyId "alias/my-key"
```

Untuk mengenkripsi ulang dengan kunci yang dikelola pelanggan yang berbeda

Gunakan Copy-EC2Volume cmdlet dengan `-KmsKeyId` parameter untuk menentukan kunci yang berbeda.

```
Copy-EC2Volume `
-SourceVolumeId vol-01234567890abcdef `
-KmsKeyId "arn:aws:kms:us-east-1:222222222222:key/abcd1234-a123-456a-a12b-
a123b4cd56ef"
```

Melampirkan volume Amazon EBS ke instans Amazon EC2

Anda dapat melampirkan volume EBS yang tersedia pada satu atau beberapa instans yang berada dalam Zona Ketersediaan yang sama dengan volume tersebut.

Untuk informasi tentang menambahkan volume EBS ke instans saat diluncurkan, lihat pemetaan perangkat blok instans.

Pertimbangan-pertimbangan

- Jumlah maksimum volume Amazon EBS yang dapat Anda lampirkan ke instans tergantung pada jenis instans. Jika Anda melebihi batas lampiran volume untuk jenis instans, permintaan lampiran gagal dengan `AttachmentLimitExceeded` kesalahan. Untuk informasi selengkapnya, lihat Batas volume instans.
- Anda dapat melampirkan volume ke instans yang hanya berada di Zona Ketersediaan yang sama.
- Multi-Attach volume yang diaktifkan dapat dilampirkan hingga 16 instance. Untuk informasi selengkapnya, lihat Melampirkan volume EBS ke beberapa instans EC2 menggunakan Multi-Attach.
- Jika volume memiliki kode AWS Marketplace produk:

  - Anda dapat melampirkannya ke instance yang dihentikan saja.
  - Anda harus berlangganan AWS Marketplace kode yang ada di volume.
  - Konfigurasi instance, seperti jenis dan sistem operasinya, harus mendukung AWS Marketplace kode spesifik tersebut. Misalnya, Anda tidak dapat mengambil volume dari instans Windows dan menempelkannya ke instans Linux.
  - AWS Marketplace kode disalin dari volume ke instance.
- Nama perangkat yang Anda tentukan ini digunakan oleh Amazon EC2. Driver perangkat blok dapat memasang perangkat dengan nama perangkat yang berbeda dari yang Anda tentukan. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume pada instans Amazon EC2.
- Dalam beberapa kasus, volume selain volume yang melekat pada `/dev/xvda` atau `/dev/sda` dapat menjadi volume root untuk instance. Hal ini dapat terjadi jika Anda melampirkan volume root dari instance lain, atau volume yang dibuat dari snapshot volume root, ke instance dengan volume root yang ada. Untuk informasi selengkapnya, lihat Boot dari volume yang salah.
- Beberapa jenis instans mendukung lebih dari satu kartu EBS. Anda dapat memilih kartu EBS untuk volume yang akan dilampirkan dengan menentukan indeks kartu EBS. Untuk contoh mendukung beberapa kartu EBS, lihat kartu EBS.
  - Volume root Anda harus dilampirkan ke indeks `0` kartu EBS.
  - Untuk contoh yang mendukung beberapa kartu EBS, Jika Anda tidak menentukan indeks kartu EBS, volume Anda akan dilampirkan ke indeks kartu EBS. `0`
  - Saat mengonfigurasi instans EC2 Anda untuk beban kerja berkinerja tinggi, penting untuk menyeimbangkan volume EBS di seluruh kartu EBS berdasarkan persyaratan kinerja, untuk menghindari terjadinya batasan kinerja pada kartu EBS mana pun.
  - Batas lampiran volume untuk jenis instans tersebar merata di setiap kartu EBS. Misalnya, pada instans EC2 yang mendukung lampiran `128` volume dengan 2 kartu EBS, setiap kartu EBS dapat mendukung hingga `64` lampiran volume. Jika Anda melebihi batas lampiran kartu EBS, permintaan gagal dengan `CardAttachmentLimitExceeded` kesalahan.

Console

Untuk melampirkan volume EBS ke instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume yang akan dilampirkan dan pilih Tindakan, Lampirkan volume.

4. Untuk Instans, masukkan ID instans atau pilih instans dari daftar opsi.
5. Untuk Nama Perangkat, lakukan salah satu hal berikut:
   - Untuk volume root, pilih nama perangkat yang diperlukan dari bagian Dicadangkan untuk volume root dari daftar. Biasanya `/dev/sda1` atau `/dev/xvda` untuk instance Linux tergantung pada AMI, atau `/dev/sda1` untuk instans Windows.
   - Untuk volume data, pilih nama perangkat yang tersedia dari bagian Direkomendasikan untuk volume data dalam daftar.
   - Untuk menggunakan nama perangkat khusus, pilih T entukan nama perangkat khusus, lalu masukkan nama perangkat yang akan digunakan.
6. Pilih Lampirkan volume.
7. Sambungkan ke instans dan pasang volume. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

AWS CLI

Untuk melampirkan volume EBS ke instans

Gunakan perintah attach-volume. Contoh berikut melampirkan volume yang ditentukan ke instance yang ditentukan menggunakan nama perangkat yang ditentukan.

```
aws ec2 attach-volume \
    --volume-id vol-01234567890abcdef \
    --instance-id i-1234567890abcdef0 \
    --device /dev/sdf
```

PowerShell

Untuk melampirkan volume EBS ke instans

Gunakan Add-EC2Volume cmdlet. Contoh berikut melampirkan volume yang ditentukan ke instance yang ditentukan menggunakan nama perangkat yang ditentukan.

```
Add-EC2Volume `
    -VolumeId vol-01234567890abcdef `
    -InstanceId i-1234567890abcdef0 `
    -Device /dev/sdf
```

Melampirkan volume EBS ke beberapa instans EC2 menggunakan Multi-Attach

Amazon EBS Multi-Attach memungkinkan Anda untuk melampirkan satu volume SSD (`io1`atau`io2`) IOPS yang disediakan ke beberapa instans yang berada di Zona Ketersediaan yang sama. Anda dapat melampirkan beberapa volume yang Multi-Attach diaktifkan ke instans atau kumpulan instance. Setiap instance yang dilampirkan volume memiliki izin baca dan tulis penuh ke volume bersama. Multi-Attachmemudahkan Anda untuk mencapai ketersediaan aplikasi yang lebih tinggi dalam aplikasi yang mengelola operasi penulisan bersamaan.

Harga dan penagihan

Tidak ada biaya tambahan untuk menggunakan Amazon EBS Multi-Attach. Anda dikenai biaya dengan tarif standar yang berlaku untuk volume SSD IOPS yang Tersedia (`io1` dan `io2`). Untuk informasi selengkapnya, lihat harga Amazon EBS.

Daftar Isi

- Pertimbangan dan batasan
- Kinerja untuk Multi-Attach volume Amazon EBS
- Akti Multi-Attach fkan untuk volume Amazon EBS
- N Multi-Attach onaktifkan untuk volume Amazon EBS
- Gunakan reservasi NVMe dengan volume Amazon EBS yang Multi-Attach diaktifkan

Pertimbangan dan batasan

- Multi-Attach volume yang diaktifkan dapat dilampirkan hingga 16 instans yang dibangun pada Sistem Nitro yang berada di Zona Ketersediaan yang sama.
- Instans Linux mendukung Multi-Attach diaktifkan `io1` dan `io2` volume. Instans Windows hanya mendukung `io2` volume yang Multi-Attach diaktifkan.
- Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Untuk informasi selengkapnya, lihat batas volume instans.
- Multi-Attach didukung secara eksklusif pada volume SSD (`io1`dan`io2`) IOPS yang disediakan.
- Multi-Attach untuk `io1` volume tersedia di semua Wilayah yang mendukung`io1`.

  Multi-Attach untuk `io2` tersedia di semua Wilayah yang mendukung`io2`.

> **Note**
>
> Untuk performa, konsistensi, dan daya tahan yang lebih baik dengan biaya lebih rendah, kami sarankan Anda menggunakan volume `io2`.

- Sistem file standar, seperti XFS dan EXT4, tidak dirancang untuk diakses secara bersamaan oleh beberapa server, seperti instans EC2. Anda harus menggunakan sistem file berkelompok untuk memastikan bahwa data Anda tangguh dan dapat diandalkan untuk beban kerja produksi Anda.
- Multi-Attach diaktifkan `io1` dan `io2` volume mendukung I/O pagar. I/O protokol pagar mengontrol akses tulis di lingkungan penyimpanan bersama untuk menjaga konsistensi data. Aplikasi Anda harus memberikan urutan penulisan untuk instans terlampir untuk menjaga konsistensi data. Untuk informasi selengkapnya, lihat Gunakan reservasi NVMe dengan volume Amazon EBS yang Multi-Attach diaktifkan.
- Multi-Attach volume yang diaktifkan tidak dapat dibuat sebagai volume boot.
- Multi-Attach volume yang diaktifkan dapat dilampirkan ke satu pemetaan perangkat blok per instance.
- Multi-Attach tidak dapat diaktifkan selama peluncuran instance menggunakan konsol atau RunInstances API Amazon EC2.
- Multi-Attach volume yang diaktifkan yang memiliki masalah di lapisan infrastruktur Amazon EBS tidak tersedia untuk semua instans terlampir. Masalah di Amazon EC2 atau lapisan jaringan mungkin hanya berdampak pada beberapa instans yang terpasang.
- Tabel berikut menunjukkan dukungan modifikasi volume untuk Multi-Attach diaktifkan `io1` dan `io2` volume setelah pembuatan.

	Volume **`io2`**	Volume **`io1`**
Mengubah tipe volume	✓ **	✓ **
Mengubah ukuran volume	✓	✓
Mengubah IOPS yang tersedia	✓	✓

	Volume **io2**	Volume **io1**
Aktifkan Multi-Attach	✓ *	✓ *
Nonaktifkan Multi-Attach	✓ *	✓ *

* Anda tidak dapat mengaktifkan atau menonaktifkan Multi-Attach saat volume dilampirkan ke instance.

** Modifikasi jenis volume untuk volume yang Multi-Attach diaktifkan hanya didukung antara `io1` dan`io2`, di kedua arah, termasuk saat volume terpasang. Perubahan jenis volume lainnya tidak didukung.

- Multi-Attach volume yang diaktifkan dihapus pada penghentian instans jika instance terlampir terakhir dihentikan dan jika instance tersebut dikonfigurasi untuk menghapus volume pada penghentian. Jika volume terlampir ke banyak instans yang memiliki pengaturan pengakhiran saat pengakhiran yang berbeda dalam pemetaan perangkat blok volumenya, pengaturan pemetaan perangkat blok instans terakhir yang terlampir menentukan penghapusan pada perilaku pengakhiran.

  Untuk memastikan penghapusan yang dapat diprediksi pada perilaku pengakhiran, aktifkan atau nonaktifkan penghapusan pada saat pengakhiran untuk semua instans tempat volume terpasang. Untuk informasi selengkapnya, lihat Memper tahankan data saat instance dihentikan.
- Anda dapat memantau volume yang Multi-Attach diaktifkan menggunakan CloudWatch Metrik untuk volume Amazon EBS. Data digabungkan di semua instans yang terlampir. Anda tidak dapat memantau metrik untuk setiap instans yang terlampir. Untuk informasi selengkapnya, lihat CloudWatch Metrik Amazon untuk Amazon EBS.

Kinerja untuk Multi-Attach volume Amazon EBS

Setiap instans yang dilampirkan mampu mendorong performa IOPS maksimum hingga performa maksimal yang tersedia dari volume. Namun, performa agregat dari semua instans yang terlampir tidak dapat melebihi performa maksimal yang tersedia dari volume. Jika permintaan instans yang terpasang untuk IOPS lebih tinggi dari volume IOPS yang Tersedia, volumenya tidak akan melebihi performa yang disediakan.

Misalnya, Anda membuat volume yang `io2` Multi-Attach diaktifkan dengan IOPS `80,000` yang disediakan dan Anda melampirkannya ke `m7g.large` instance yang mendukung hingga `40,000` IOPS, dan `r7g.12xlarge` instance yang mendukung hingga IOPS. `60,000` Setiap instans dapat mendorong IOPS maksimum karena kurang dari volume IOPS yang tersedia sebesar `80,000`. Namun, jika kedua instans mengar I/O ahkan ke volume secara bersamaan, IOPS gabungannya tidak dapat melebihi kinerja IOPS yang disediakan volume. `80,000`

Untuk mencapai kinerja yang konsisten, praktik terbaik adalah menyeimbangkan dor I/O ongan dari instance terlampir di seluruh sektor volume yang Multi-Attach diaktifkan.

Untuk informasi selengkapnya tentang kinerja IOPS untuk jenis instans Amazon EC2, lihat jenis instans yang dioptimalkan Amazon EBS di Panduan Pengguna Amazon EC2.

Akti Multi-Attach fkan untuk volume Amazon EBS

Multi-Attach volume yang diaktifkan dapat dikelola dengan cara yang sama seperti Anda mengelola volume Amazon EBS lainnya. Namun, untuk menggunakan Multi-Attach fungsionalitas, Anda harus mengaktifkannya untuk volume.

Saat Anda membuat volume baru, dinon Multi-Attach aktifkan secara default. Anda dapat mengaktifkan Multi-Attach saat Anda membuat volume.

Anda juga dapat mengaktifkan Multi-Attach for `io1` dan `io2` volume setelah pembuatan, tetapi hanya jika mereka tidak dilampirkan ke instance apa pun.

Setelah Anda mengaktifkan Multi-Attach volume, Anda dapat melampirkan volume ke instance dengan cara yang sama seperti Anda melampirkan volume EBS lainnya. Untuk informasi selengkapnya, lihat Melampirkan volume Amazon EBS ke instans Amazon EC2.

Console

Untuk mengaktifkan Multi-Attach selama pembuatan volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih Buat Volume.
4. Untuk Tipe volume, pilih SSD IOPS yang Tersedia (**io1**) atau SSD IOPS yang Tersedia (**io2**).
5. Untuk Ukuran dan IOPS, pilih ukuran volume yang diperlukan dan jumlah IOPS untuk disediakan.

6. Untuk Zona Ketersediaan, pilih Zona Ketersediaan yang sama dengan lokasi instans.
7. Untuk Amazon EBS Multi-Attach, pilih Akti Multi-Attach fkan.
8. (Opsional) Untuk ID Snapshot, pilih snapshot tempat pembuatan volume.
9. Atur status enkripsi untuk volume.

   Jika snapshot yang dipilih dienkripsi, atau jika akun Anda diaktifkan untuk enkripsi secara default, enkripsi diaktifkan secara otomatis dan Anda tidak dapat menonaktifkannya. Anda dapat memilih kunci KMS untuk mengenkripsi volume.

   Jika snapshot yang dipilih tidak dienkripsi dan akun Anda tidak diaktifkan untuk enkripsi secara default, enkripsi bersifat opsional. Untuk mengenkripsi volume, untuk Enkripsi, pilih Enkripsi volume ini lalu pilih kunci KMS yang akan digunakan untuk mengenkripsi volume.

   Volume yang dienkripsi hanya dapat dilampirkan ke instans yang mendukung enkripsi Amazon EBS. Untuk informasi selengkapnya, lihat Enkripsi EBS Amazon.
10. (Opsional) Untuk menetapkan tanda kustom ke volume, di bagian Tanda, pilih Tambahkan tanda, lalu masukkan kunci tanda dan pasangan nilai.
11. Pilih Buat volume.

Untuk mengaktifkan Multi-Attach setelah pembuatan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume dan pilih Tindakan, Ubah Volume.
4. Untuk Amazon EBS Multi-Attach, pilih Akti Multi-Attach fkan.
5. Pilih Ubah.

AWS CLI

Untuk mengaktifkan Multi-Attach selama pembuatan volume

Gunakan perintah create-volume dengan opsi. `--multi-attach-enabled`

```
aws ec2 create-volume \
    --volume-type io2 \
    --multi-attach-enabled \
```

```
    --size 100 \
    --iops 2000 \
    --region us-west-2 \
    --availability-zone us-west-2b
```

Untuk mengaktifkan Multi-Attach setelah pembuatan

Gunakan perintah modify-volume dengan opsi. `--multi-attach-enabled`

```
aws ec2 modify-volume \
    --volume-id vol-01234567890abcdef \
    --multi-attach-enabled
```

PowerShell

Untuk mengaktifkan Multi-Attach selama pembuatan volume

Gunakan New-EC2Volume cmdlet dengan parameter. `-MultiAttachEnabled`

```
New-EC2Volume `
    -VolumeType io2 `
    -MultiAttachEnabled $true `
    -Size 100 `
    -Iops 2000 `
    -Region us-west-2 `
    -AvailabilityZone us-west-2b
```

Untuk mengaktifkan Multi-Attach setelah pembuatan

Gunakan Edit-EC2Volume cmdlet dengan parameter. `-MultiAttachEnabled`

```
Edit-EC2Volume `
    -VolumeId vol-01234567890abcdef `
    -MultiAttachEnabled $true
```

N Multi-Attach onaktifkan untuk volume Amazon EBS

Anda dapat menonaktifkan Multi-Attach untuk `io2` volume `io1` atau hanya jika tidak dilampirkan ke instance.

Console

Untuk menonaktifkan Multi-Attach setelah pembuatan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume dan pilih Tindakan, Ubah Volume.
4. Untuk Amazon EBS Multi-Attach, hapus Akti Multi-Attach fkan.
5. Pilih Ubah.

AWS CLI

Untuk menonaktifkan Multi-Attach setelah pembuatan

Gunakan perintah modify-volume dengan opsi. `-no-multi-attach-enabled`

```
aws ec2 modify-volume \
    --volume-id vol-01234567890abcdef \
    --no-multi-attach-enabled
```

PowerShell

Untuk menonaktifkan Multi-Attach setelah pembuatan

Gunakan Edit-EC2Volume cmdlet dengan parameter. `-MultiAttachEnabled`

```
Edit-EC2Volume `
    -VolumeId vol-01234567890abcdef `
    -MultiAttachEnabled $false
```

Gunakan reservasi NVMe dengan volume Amazon EBS yang Multi-Attach diaktifkan

Multi-Attach diaktifkan `io1` dan `io2` volume mendukung reservasi NVMe, yang merupakan seperangkat protokol pagar penyimpanan standar industri. Protokol ini memungkinkan Anda membuat dan mengelola reservasi yang mengontrol dan mengoordinasikan akses dari beberapa instans ke volume bersama. Aplikasi penyimpanan bersama menggunakan reservasi untuk memastikan bahwa data tetap konsisten.

Daftar Isi

- Persyaratan
- Mengaktifkan dukungan untuk reservasi NVMe
- Perintah Reservasi NVMe yang didukung
- Harga

Persyaratan

Reservasi NVMe didukung dengan Multi-Attach diaktifkan `io1` dan `io2` volume. Multi-Attach volume yang diaktifkan hanya dapat dilampirkan ke instance yang dibangun pada sistem Nitro.

Reservasi NVMe didukung dengan sistem operasi berikut ini:

- SUSE Linux Enterprise 12 SP3 dan lebih baru
- RHEL 8.3 dan yang lebih baru
- Amazon Linux 2 dan yang lebih baru
- Windows Server 2016 dan setelahnya

> **Note**
>
> Untuk AMI Windows Server yang didukung tertanggal 2023.09.13 dan yang lebih baru, driver NVMe yang diperlukan disertakan. Untuk AMI sebelumnya, Anda harus memperbarui ke driver NVMe versi 1.5.0 atau yang lebih baru. Untuk informasi selengkapnya, lihat driver AWS NVMe.

Jika Anda menggunakan EC2launch v2 untuk menginisialisasi disk, Anda harus memutakhirkan ke versi 2. 0.1521 atau yang lebih baru. Untuk informasi selengkapnya, lihat Menggunakan agen EC2launch v2.

Mengaktifkan dukungan untuk reservasi NVMe

Dukungan untuk reservasi NVMe diaktifkan secara default untuk semua `io2` volume Multi-Attach yang diaktifkan yang dibuat setelah 18 September 2023, dan untuk semua `io1` volume yang Multi-Attach diaktifkan yang dibuat setelah 1 Oktober 202 6.

Untuk mengaktifkan dukungan reservasi NVMe untuk `io2` volume yang ada yang dibuat sebelum 18 September 2023, atau `io1` volume yang ada yang dibuat sebelum 1 Oktober 2026, Anda harus

melepaskan semua instans dari volume dan kemudian memasang kembali instance yang diperlukan. Semua lampiran yang dibuat setelah melepaskan lampiran semua instans akan mengaktifkan reservasi NVMe.

Perintah Reservasi NVMe yang didukung

Amazon EBS mendukung perintah Reservasi NVMe berikut:

Registrasi Reservasi

Mendaftarkan, membatalkan pendaftaran, atau mengganti kunci reservasi. Kunci registrasi digunakan untuk mengidentifikasi dan mengautentikasi sebuah instans. Mendaftarkan kunci reservasi dengan volume menciptakan kairan antara instans dan volume. Anda harus mendaftarkan instans dengan volume sebelum instans itu dapat memperoleh reservasi.

Pemerolehan Reservasi

Memperoleh reservasi pada volume, mendahului reservasi yang disimpan di namespace, dan membatalkan reservasi yang disimpan pada volume. Jenis reservasi berikut dapat diperoleh:

- Tulis Reservasi Eksklusif
- Reservasi Akses Eksklusif
- Tulis Eksklusif - Hanya Reservasi Pendaftar
- Akses Eksklusif - Reservasi Khusus Pendaftar
- Tulis Eksklusif - Reservasi Semua Pendaftar
- Akses Eksklusif - Reservasi Semua Pendaftar

Rilis Reservasi

Merilis atau menghapus reservasi yang disimpan pada volume.

Laporan Reservasi

Menjelaskan status pendaftaran dan reservasi volume.

Harga

Tidak ada biaya tambahan untuk mengaktifkan dan menggunakan Multi-Attach.

Menyediakan volume Amazon EBS untuk digunakan

Setelah Anda melampirkan volume Amazon EBS ke instans, volume tersebut akan ditampilkan sebagai perangkat blok. Anda dapat memformat volume dengan sembarang sistem file

lalu memasangnya. Setelah Anda menyediakan volume EBS untuk digunakan, Anda dapat mengaksesnya dengan cara yang sama seperti Anda mengakses volume lainnya. Setiap data yang ditulis pada sistem file ini ditulis ke volume EBS dan terlihat untuk aplikasi yang menggunakan perangkat tersebut.

Anda dapat mengambil foto volume EBS untuk tujuan pencadangan atau menggunakannya sebagai dasar saat Anda membuat volume lain. Untuk informasi selengkapnya, lihat Snapshot Amazon EBS.

Jika volume EBS yang Anda persiapkan untuk digunakan lebih besar dari 2 TiB, Anda harus menggunakan skema partisi GPT untuk mengakses seluruh volume. Untuk informasi selengkapnya, lihat Batasan volume Amazon EBS.

Instans Linux

Format dan pasang volume yang terpasang

Misalkan, Anda memiliki instans EC2 dengan volume EBS untuk perangkat root, `/dev/xvda`, dan bahwa Anda baru saja melampirkan volume EBS kosong pada instans menggunakan `/dev/sdf`. Gunakan prosedur berikut untuk membuat volume baru terpasang tersedia untuk digunakan.

Untuk memformat dan memasang volume EBS di Linux

1. Connect ke instans Anda dengan menggunakan SSH. Untuk informasi selengkapnya, lihat Menghubungkan ke instance Linux Anda.
2. Perangkat dapat dilampirkan ke instans dengan nama perangkat yang berbeda dengan yang Anda tentukan dalam pemetaan perangkat blok. Untuk informasi selengkapnya, lihat nama perangkat pada instans Linux. Gunakan perintah lsblk untuk melihat perangkat disk yang tersedia dan titik pemasangannya (jika ada) untuk membantu Anda menentukan nama perangkat yang tepat untuk digunakan. Output dari lsblk menghapus prefiks `/dev/` dari jalur perangkat lengkap.

   Berikut ini adalah contoh output untuk instance yang dibangun pada Sistem Nitro, yang mengekspos volume EBS sebagai perangkat blok NVMe. Perangkat root adalah `/dev/nvme0n1`, yang memiliki dua partisi bernama `nvme0n1p1` dan `nvme0n1p128`. Volume yang terlampir adalah `/dev/nvme1n1`, yang tidak memiliki partisi dan belum dipasang.

```
[ec2-user ~]$ lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme1n1       259:0    0  10G  0 disk
nvme0n1       259:1    0   8G  0 disk
```

```
-nvme0n1p1    259:2    0   8G  0 part /
-nvme0n1p128  259:3    0   1M  0 part
```

Berikut ini instans output untuk instans T2. Perangkat root adalah `/dev/xvda`, yang memiliki satu partisi bernama `xvda1`. Volume yang terlampir adalah `/dev/xvdf`, yang tidak memiliki partisi dan belum dipasang.

```
[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   10G  0 disk
```

3. Tentukan apakah ada sistem file pada volume. Volume baru adalah perangkat blok mentah, dan Anda harus membuat sistem file di dalamnya sebelum Anda dapat memasang dan menggunakannya. Volume yang dibuat dari snapshot mungkin sudah memiliki sistem file; jika Anda membuat sistem file baru di atas sistem file yang sudah ada, operasi akan menimpa data Anda.

   Gunakan salah satu atau kedua metode berikut untuk menentukan apakah ada sistem file pada volume:

   - Gunakan perintah file -s untuk mendapatkan informasi tentang perangkat spesifik, seperti tipe sistem file-nya. Jika output menunjukkan hanya `data`, seperti pada contoh output berikut, tidak ada sistem file di perangkat

     ```
     [ec2-user ~]$ sudo file -s /dev/xvdf
     /dev/xvdf: data
     ```

     Jika perangkat memiliki sistem file, perintah akan menampilkan informasi tentang jenis sistem file. Misalnya, output berikut menunjukkan perangkat root dengan sistem file XFS.

     ```
     [ec2-user ~]$ sudo file -s /dev/xvda1
     /dev/xvda1: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
     ```

   - Gunakan perintah lsblk -f untuk mendapatkan informasi tentang semua perangkat yang terlampir pada instans.

     ```
     [ec2-user ~]$ sudo lsblk -f
     ```

Misalnya, output berikut menunjukkan bahwa ada tiga perangkat yang dilampirkan ke instans—nvme1n1, nvme0n1, dan nvme2n1. Kolom pertama mencantumkan perangkat dan partisi mereka. Kolom FSTYPE menunjukkan jenis sistem file untuk setiap perangkat. Jika kolom kosong untuk perangkat tertentu, itu berarti perangkat tidak memiliki sistem file. Dalam hal ini, perangkat nvme1n1 dan partisi nvme0n1p1 pada perangkat nvme0n1 keduanya diformat menggunakan sistem file XFS, sedangkan perangkat nvme2n1 dan partisi nvme0n1p128 pada perangkat nvme0n1 tidak memiliki sistem file.

```
NAME  FSTYPE LABEL UUID      MOUNTPOINT
nvme1n1         xfs  7f939f28-6dcc-4315-8c42-6806080b94dd
nvme0n1
##nvme0n1p1 xfs     / 90e29211-2de8-4967-b0fb-16f51a6e464c         /
##nvme0n1p128
nvme2n1
```

Jika output dari perintah ini menunjukkan bahwa tidak ada sistem file pada perangkat, Anda harus membuatnya.

4. (Bersyarat) Jika Anda menemukan bahwa ada sistem file pada perangkat di langkah sebelumnya, lewati langkah ini. Jika Anda memiliki volume kosong, gunakan perintah mkfs -t untuk membuat sistem file pada volume.

> **Warning**
>
> Jangan gunakan perintah ini jika Anda memasang volume yang sudah memiliki data di dalamnya (misalnya, volume yang dibuat dari snapshot). Jika tidak, Anda akan memformat volume dan menghapus data yang ada.

```
[ec2-user ~]$ sudo mkfs -t xfs /dev/xvdf
```

Jika Anda menerima pesan kesalahan bahwa mkfs.xfs tidak ditemukan, gunakan perintah berikut untuk menginstal alat XFS, lalu ulangi perintah sebelumnya:

```
[ec2-user ~]$ sudo yum install xfsprogs
```

5. Gunakan perintah mkdir untuk membuat direktori titik pasang untuk volume. Titik pasang adalah tempat volume berada di struktur sistem file dan tempat Anda membaca serta menulis file setelah Anda memasang volume. Contoh berikut membuat direktori yang bernama `/data`.

   ```
   [ec2-user ~]$ sudo mkdir /data
   ```

6. Pasang volume atau partisi pada direktori titik pemasangan yang Anda buat pada langkah sebelumnya.

   Jika volume tidak memiliki partisi, gunakan perintah berikut dan tentukan nama perangkat untuk memasang seluruh volume.

   ```
   [ec2-user ~]$ sudo mount /dev/xvdf /data
   ```

   Jika volume memiliki partisi, gunakan perintah berikut dan tentukan nama partisi untuk memasang partisi.

   ```
   [ec2-user ~]$ sudo mount /dev/xvdf1 /data
   ```

7. Tinjaulah izin file untuk pemasangan volume baru Anda untuk memastikan bahwa pengguna dan aplikasi Anda dapat menulis ke volume. Untuk informasi selengkapnya tentang izin file, lihat Keamanan file di Proyek Dokumentasi Linux.
8. Titik pemasangan tidak dipertahankan secara otomatis setelah melakukan boot ulang instans Anda. Untuk memasang volume EBS ini secara otomatis setelah reboot, ikuti prosedur berikutnya.

Otomatis memasang volume yang terlampir setelah boot ulang

Untuk memasang volume EBS yang terlampir pada setiap boot ulang sistem, tambahkan entri untuk perangkat ke file `/etc/fstab` Anda.

Anda dapat menggunakan nama perangkat, seperti `/dev/xvdf`, di `/etc/fstab`, tetapi sebaiknya gunakan pengidentifikasi yang unik universal (UUID) 128-bit. Nama perangkat dapat berubah, tetapi UUID tetap ada selama masa paruh. Dengan menggunakan UUID, Anda mengurangi kemungkinan sistem menjadi tidak dapat diaktifkan setelah konfigurasi ulang perangkat keras. Untuk informasi selengkapnya, lihat Petakan volume Amazon EBS ke nama perangkat NVMe.

Untuk otomatis memasang volume yang terlampir setelah boot ulang

1. (Opsional) Buat cadangan dari file `/etc/fstab` Anda yang dapat digunakan jika Anda secara tidak sengaja menghancurkan atau menghapus file ini saat mengedit.

   ```
   [ec2-user ~]$ sudo cp /etc/fstab /etc/fstab.orig
   ```

2. Gunakan perintah blkid untuk menemukan UUID perangkat. Buat catatan UUID perangkat yang ingin Anda pasang setelah boot ulang. Anda akan membutuhkannya dalam langkah berikut.

   Sebagai contoh, perintah berikut menunjukkan bahwa ada dua perangkat yang dipasang ke instans, dan menunjukkan UUID untuk kedua perangkat.

   ```
   [ec2-user ~]$ sudo blkid
   /dev/xvda1: LABEL="/" UUID="ca774df7-756d-4261-a3f1-76038323e572" TYPE="xfs"
    PARTLABEL="Linux" PARTUUID="02dcd367-e87c-4f2e-9a72-a3cf8f299c10"
   /dev/xvdf: UUID="aebf131c-6957-451e-8d34-ec978d9581ae" TYPE="xfs"
   ```

   Untuk Ubuntu 18.04, gunakan perintah lsblk.

   ```
   [ec2-user ~]$ sudo lsblk -o +UUID
   ```

3. Buka file `/etc/fstab` menggunakan editor teks, seperti nano atau vim.

   ```
   [ec2-user ~]$ sudo vim /etc/fstab
   ```

4. Tambahkan entri berikut ke `/etc/fstab` untuk memasang perangkat di titik pemasangan yang ditentukan. Kolom tersebut adalah nilai UUID yang dikembalikan oleh blkid (atau lsblk untuk Ubuntu 18.04), titik pemasangan, sistem file, dan opsi pemasangan sistem file yang direkomendasikan. Untuk informasi lebih lanjut tentang bidang yang diperlukan, jalankan `man fstab` Untuk membuka fstab manual.

   Contoh berikut memasang perangkat dengan UUID `aebf131c-6957-451e-8d34-ec978d9581ae` untuk mount point `/data` dan menggunakan sistem `xfs` file. Ia juga menggunakan ben `nofail` dera `defaults` dan. Ini menentukan `0` untuk mencegah sistem file dibuang, dan menentukan 2 untuk menunjukkan bahwa itu adalah perangkat non-root.

   ```
   UUID=aebf131c-6957-451e-8d34-ec978d9581ae  /data  xfs  defaults,nofail  0  2
   ```

> **Note**
>
> Jika Anda pernah melakukan boot pada instans Anda tanpa melampirkan volume ini (misalnya, setelah memindahkan volume ke instans lain), opsi pemasangan `nofail` memungkinkan instans di-boot meskipun terdapat kesalahan saat memasang volume. Derivatif Debian, termasuk versi Ubuntu yang lebih awal dari 16.04, juga harus menambahkan opsi pemasangan `nobootwait`.

5. Untuk memverifikasi bahwa entri Anda bekerja, jalankan perintah berikut untuk melepas perangkat, kemudian memasang semua sistem file di `/etc/fstab`. Jika tidak ada kesalahan, file `/etc/fstab` akan baik-baik saja dan sistem file Anda akan memasangkannya secara otomatis setelah di-boot ulang.

```
[ec2-user ~]$ sudo umount /data
[ec2-user ~]$ sudo mount -a
```

Jika Anda menerima pesan kesalahan, atasi kesalahan dalam file.

> **Warning**
>
> Kesalahan dalam file `/etc/fstab` dapat membuat sistem tidak dapat di-boot. Jangan mematikan sistem yang memiliki kesalahan di `/etc/fstab` file Anda.

Jika Anda tidak yakin bagaimana cara memperbaiki kesalahan di `/etc/fstab` dan Anda telah membuat file cadangan di langkah pertama prosedur ini, Anda dapat memulihkan dari file cadangan menggunakan perintah berikut.

```
[ec2-user ~]$ sudo mv /etc/fstab.orig /etc/fstab
```

Instans Windows

Gunakan salah satu metode berikut untuk membuat volume tersedia pada instance Windows.

PowerShell

Untuk membuat semua volume EBS dengan partisi mentah tersedia untuk digunakan dengan Windows PowerShell

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat Menghubungkan ke instans Windows Anda.
2. Pada bilah tugas, buka menu Start, dan pilih Windows PowerShell.
3. Gunakan serangkaian PowerShell perintah Windows yang disediakan dalam PowerShell prompt yang terbuka. Skrip tersebut melakukan tindakan-tindakan berikut ini secara default:
   1. Menghentikan layanan ShellHWDetection.
   2. Melakukan enumerasi disk yang gaya partisinya mentah.
   3. Membuat partisi baru yang mencakup ukuran maksimum yang akan didukung oleh disk dan jenis partisi.
   4. Menetapkan huruf drive yang tersedia.
   5. Memformat sistem file sebagai NTFS dengan label sistem file yang ditentukan.
   6. Memulai kembali layanan ShellHWDetection.

```
Stop-Service -Name ShellHWDetection
Get-Disk | Where PartitionStyle -eq 'raw' | Initialize-Disk -PartitionStyle MBR
 -PassThru | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -
FileSystem NTFS -NewFileSystemLabel "Volume Label" -Confirm:$false
Start-Service -Name ShellHWDetection
```

DiskPart command line tool

Untuk membuat volume EBS tersedia untuk digunakan dengan alat DiskPart baris perintah

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat Menghubungkan ke instans Windows Anda.
2. Tentukan nomor disk yang ingin Anda sediakan:
   1. Buka menu Start, dan pilih Windows PowerShell.
   2. Gunakan Cmdlet `Get-Disk` untuk mengambil daftar disk yang tersedia.
   3. Dalam output perintah, perhatikan Nomor yang sesuai dengan disk yang Anda sediakan.

3. Buat file skrip untuk menjalankan DiskPart perintah:

   1. Buka menu Start, dan pilih File Explorer.
   2. Arahkan ke direktori, seperti C:\, untuk menyimpan file skrip.
   3. Pilih (klik kanan) ruang kosong di dalam folder untuk membuka kotak dialog, posisikan kursor di atas Baru untuk mengakses menu konteks, lalu pilih Dokumen Teks.
   4. Beri nama file teks `diskpart.txt`.

4. Tambahkan perintah berikut ke file skrip. Anda mungkin perlu memodifikasi nomor disk, jenis partisi, label volume, dan huruf drive. Skrip tersebut melakukan tindakan-tindakan berikut ini secara default:

   1. Memilih disk 1 untuk modifikasi.
   2. Mengonfigurasi volume untuk menggunakan struktur partisi master boot record (MBR).
   3. Memformat volume sebagai volume NTFS.
   4. Mengatur label volume.
   5. Menetapkan volume huruf drive.

   > **Warning**
   >
   > Jika Anda memasang volume yang sudah memiliki data, jangan memformat ulang volume atau Anda akan menghapus data yang ada.

   ```
   select disk 1
   attributes disk clear readonly
   online disk noerr
   convert mbr
   create partition primary
   format quick fs=ntfs label="volume_label"
   assign letter="drive_letter"
   ```

   Untuk informasi selengkapnya, lihat Sintaks dan parameter DiskPart .

5. Buka prompt perintah, arahkan ke folder tempat skrip berada, dan jalankan perintah berikut agar volume tersedia untuk digunakan pada disk yang ditentukan:

```
C:\> diskpart /s diskpart.txt
```

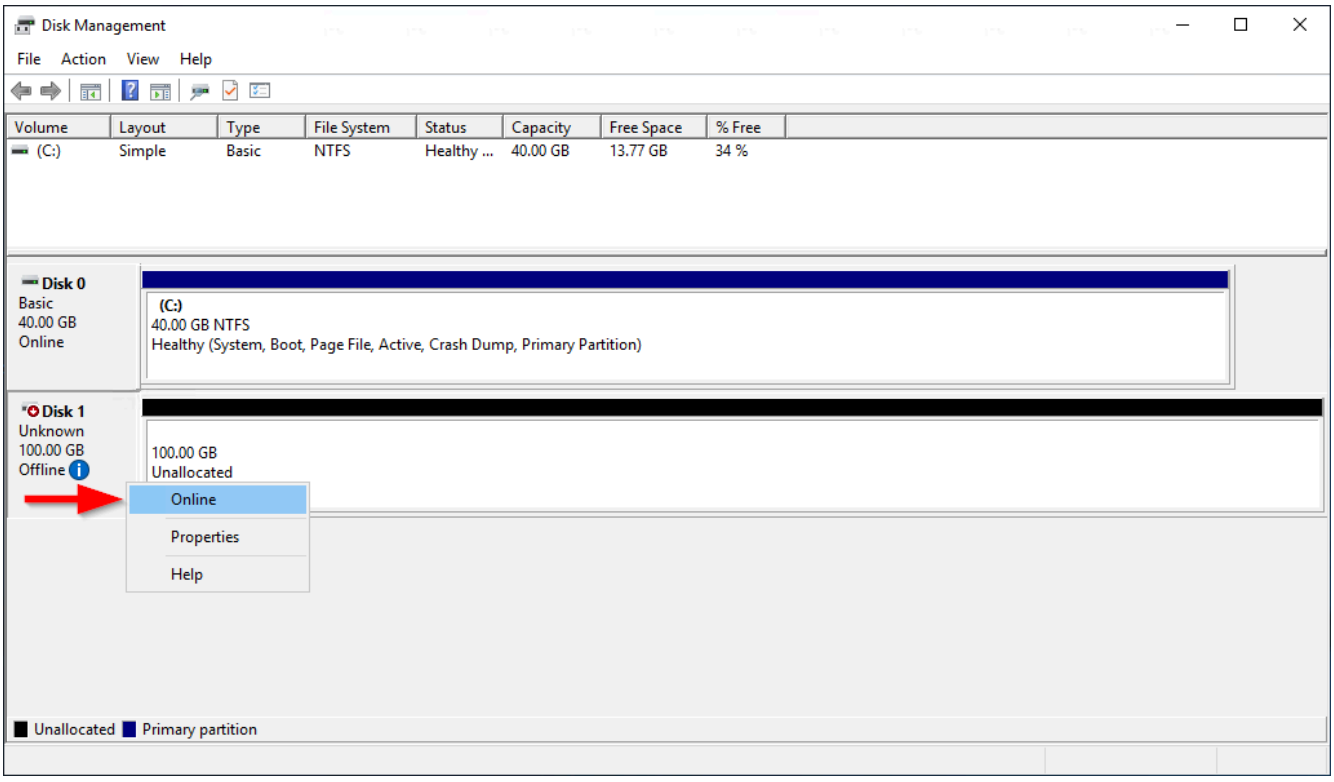
Disk Management utility

Untuk menjadikan volume EBS tersedia untuk digunakan dengan alat baris perintah DiskPart

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat Menghubungkan ke instans Windows Anda.
2. Mulai utilitas Manajemen Disk. Pada bilah tugas, buka menu konteks (klik kanan) untuk logo Windows dan pilih Manajemen Disk.

> **Note**
>
> Di Windows Server 2008, pilih Mulai, Alat Administratif, Manajemen Komputer, Manajemen Disk.

3. Buat volume menjadi online. Di panel bawah, buka menu konteks (klik kanan) untuk panel kiri untuk disk untuk volume EBS. Pilih Online.

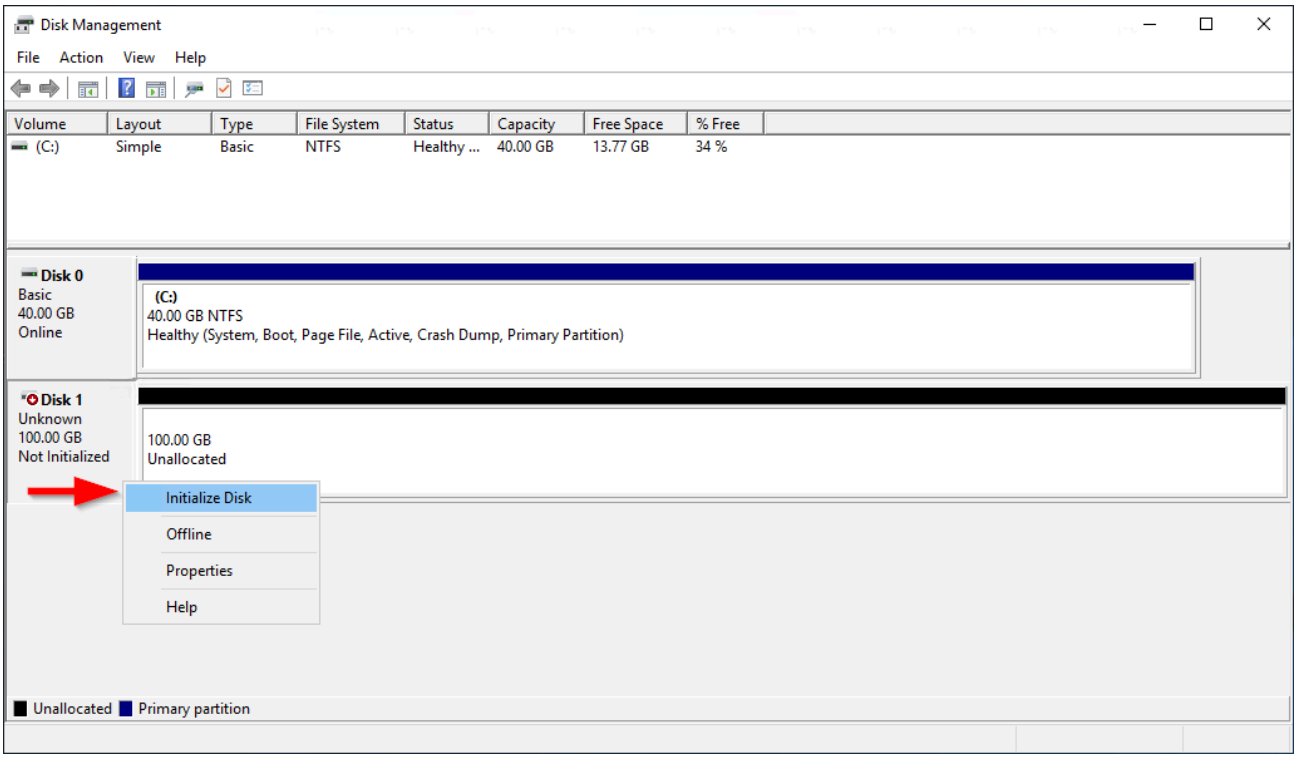


4. (Syarat) Jika disk tidak diinisialisasi, Anda harus menginisialisasinya sebelum Anda dapat menggunakannya. Jika disk sudah diinisialisasi, lewati langkah ini.

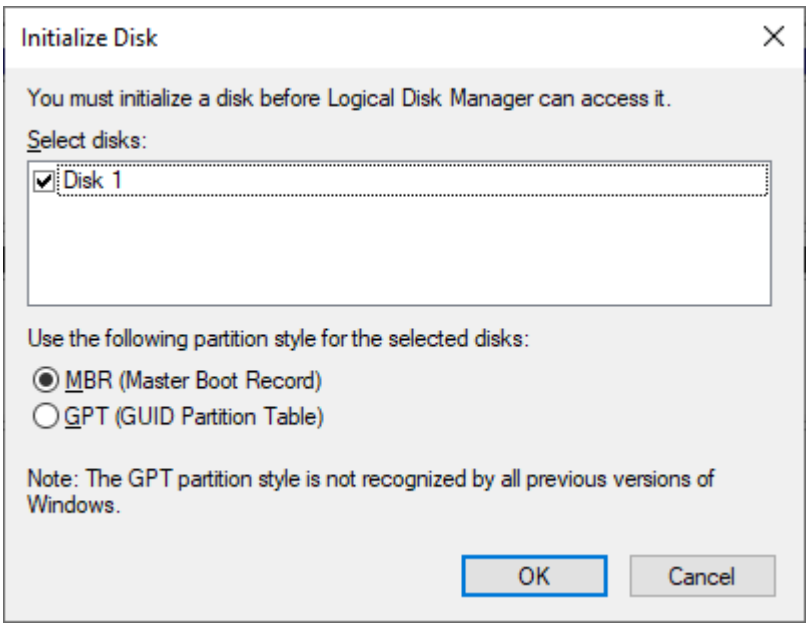
> **Warning**
>
> Jika Anda memasang volume yang sudah memiliki data di dalamnya (misalnya, set data publik, atau volume yang Anda buat dari snapshot), jangan memformat ulang volume atau data yang ada akan terhapus.

Jika disk tidak diinisialisasi, lakukan inisialisasi sebagai berikut:

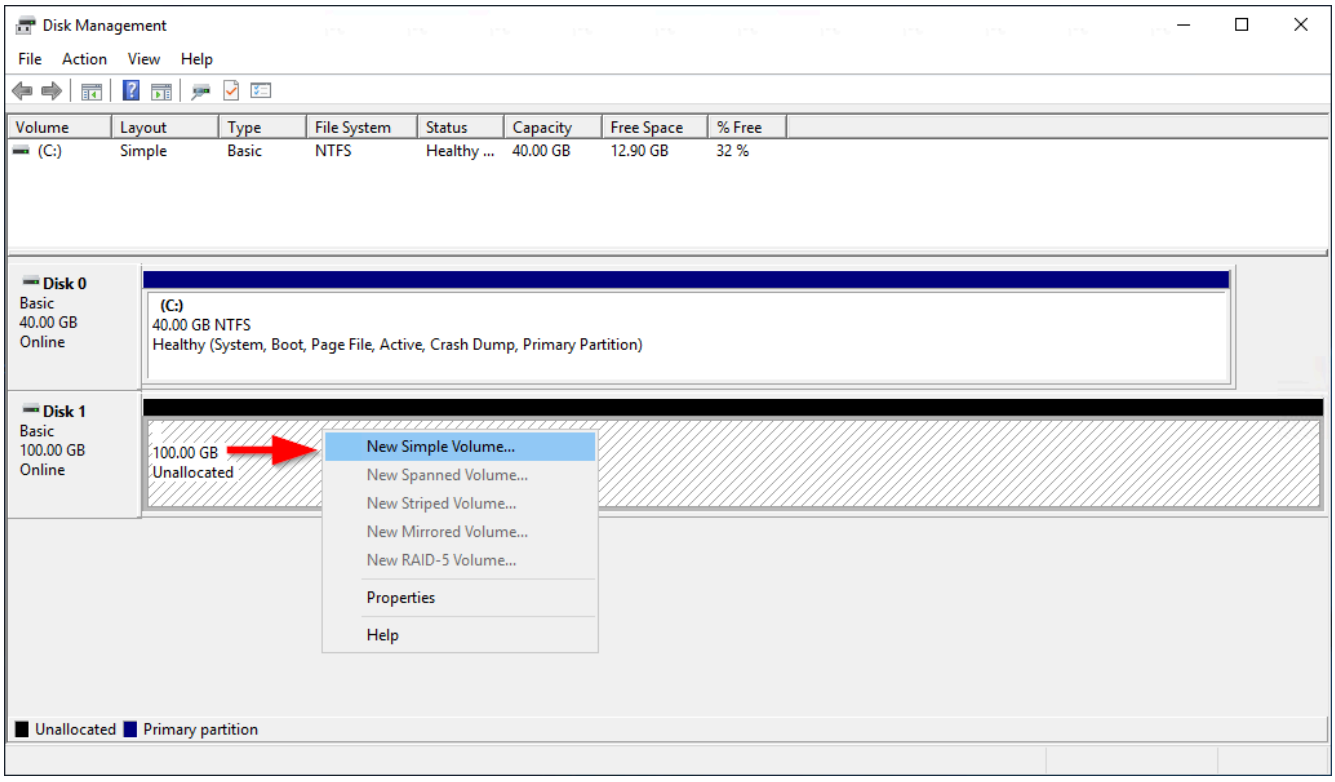
1. Buka menu konteks (klik kanan) untuk panel kiri untuk disk dan pilih Inisialisasi Disk.



2. Di kotak dialog Inisialisasi Disk, pilih gaya partisi dan pilih OKE.



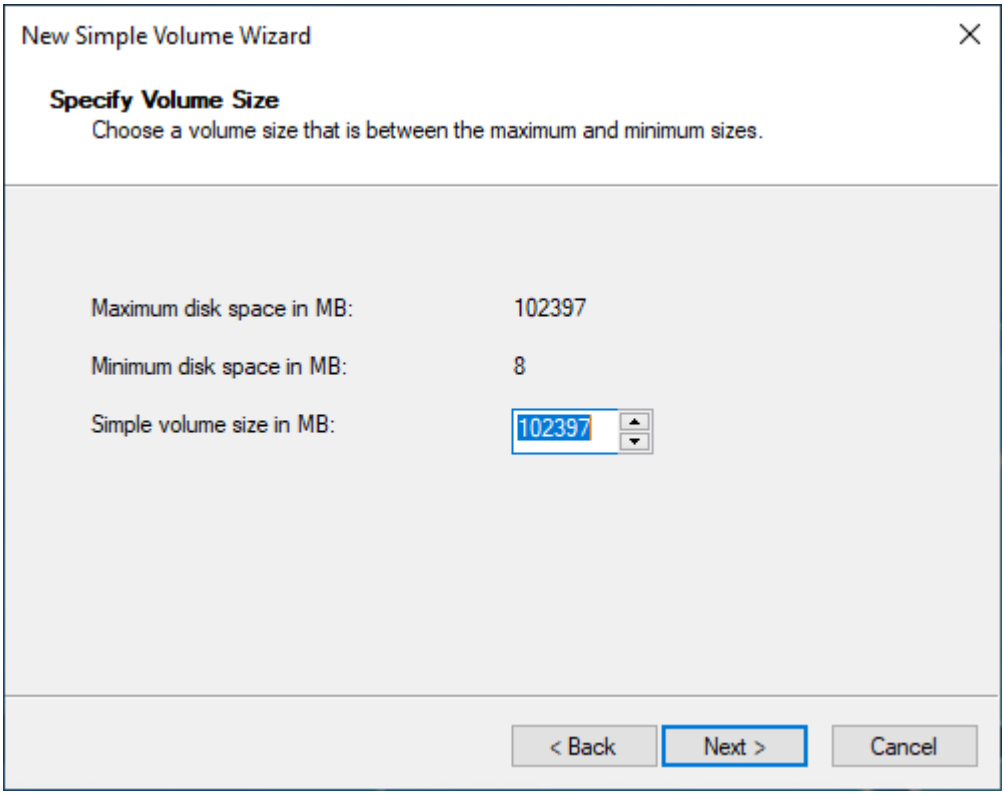
5. Buka menu konteks (klik kanan) untuk panel kanan untuk disk dan pilih Volume Sederhana Baru.



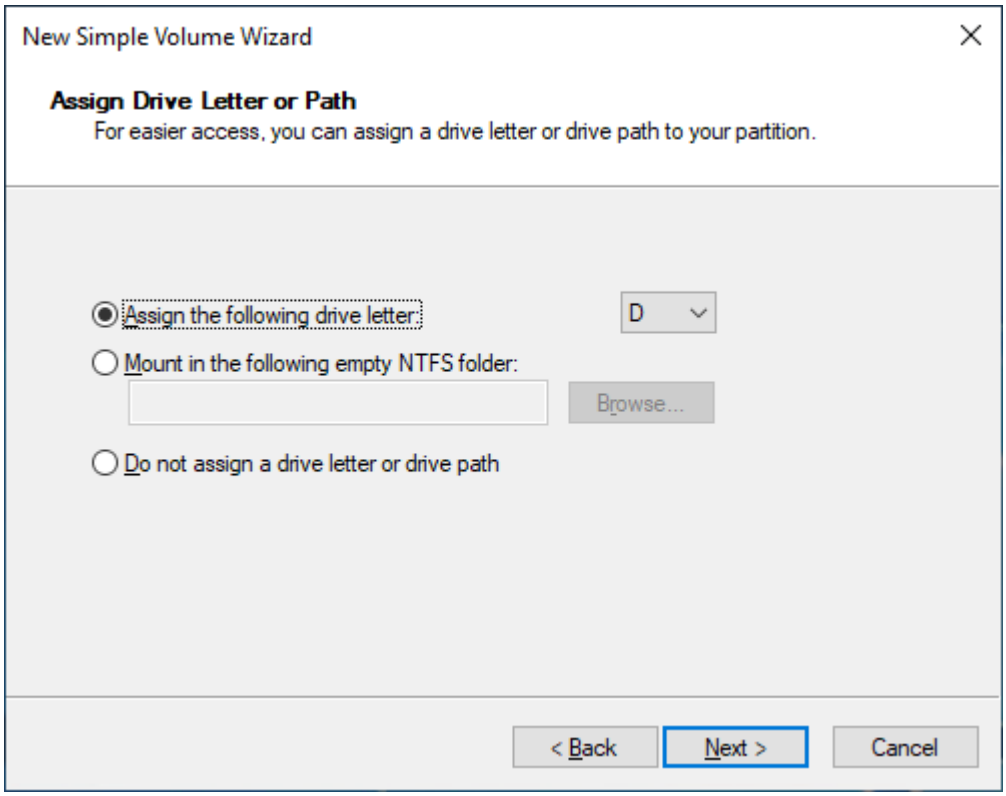
6. Di Wizard Volumer Sederhana Baru, pilih Berikutnya.



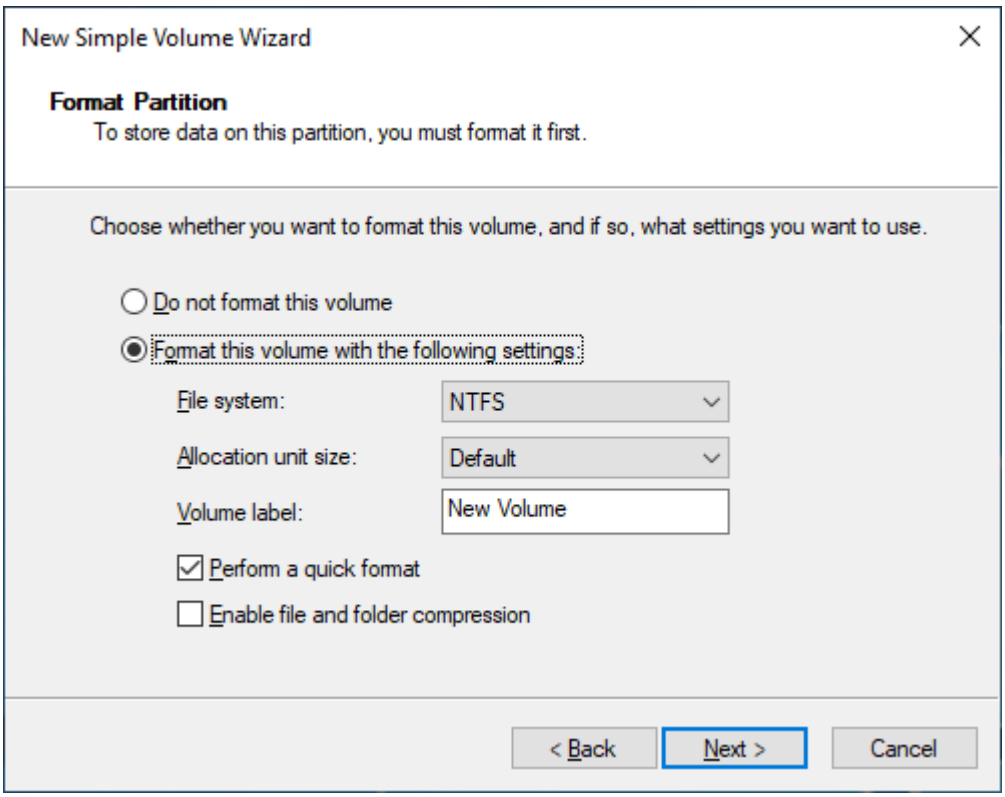
7. Jika Anda ingin mengubah nilai maksimum default, tentukan Ukuran volume sederhana dalam MB, lalu pilih Berikutnya.



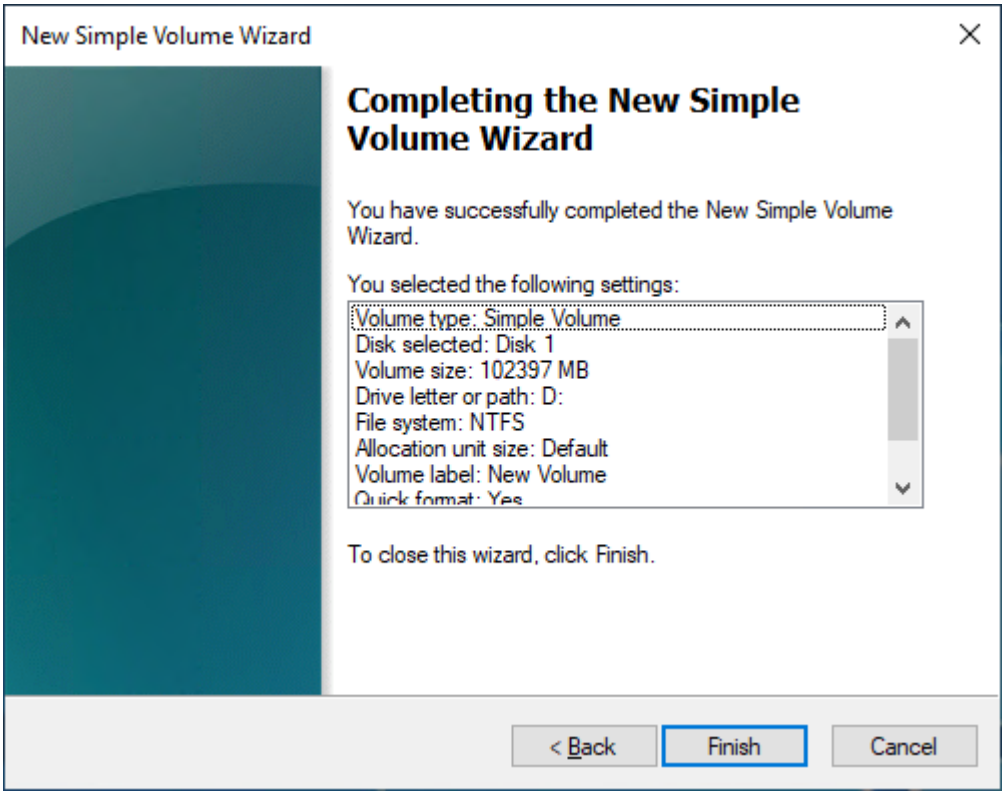
8. Tentukan huruf drive yang disukai, jika perlu, di dalam menu dropdown Tetapkan huruf drive berikut, lalu pilih Berikutnya.



9. Tentukan Label Volume dan sesuaikan pengaturan default seperlunya, lalu pilih Berikutnya.



10. Tinjau pengaturan Anda, lalu pilih Selesai untuk menerapkan modifikasi dan tutup wizard Volume Sederhana Baru.



Melihat informasi tentang volume Amazon EBS

Anda dapat melihat informasi deskriptif tentang volume EBS Anda. Misalnya, Anda dapat melihat informasi tentang semua volume di Wilayah tertentu atau melihat informasi terperinci tentang satu volume, termasuk ukurannya, jenis volume, apakah volume dienkripsi, kunci KMS mana yang digunakan untuk mengenkripsi volume, dan instance spesifik yang dilampirkan volume tersebut.

Anda dapat memperoleh informasi tambahan tentang volume EBS Anda, seperti berapa banyak ruang disk yang tersedia, dari sistem operasi pada instans.

Daftar Isi

- Lihat informasi volume
- Status volume
- Lihat metrik volume
- Lihat ruang disk kosong

Lihat informasi volume

Anda dapat melihat informasi tentang volume EBS Anda.

Console

Untuk melihat informasi tentang volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Untuk mengurangi daftar, Anda dapat memfilter volume menggunakan tanda dan atribut volume. Pilih bidang filter, pilih atribut tanda atau volume, lalu pilih nilai filter.
4. Untuk melihat informasi lebih lanjut tentang volume, pilih ID.

Untuk melihat volume EBS yang dilampirkan ke instance

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

4. Pada tab Penyimpanan, bagian Perangkat blok mencantumkan volume yang dilampirkan ke instans. Untuk melihat informasi tentang volume tertentu, pilih ID-nya di kolom ID Volume.

Amazon EC2 Global View

Anda dapat menggunakan Amazon EC2 Global View untuk melihat volume di semua Wilayah yang AWS akun Anda diaktifkan.

Untuk mendapatkan ringkasan volume EBS Anda di semua Wilayah

1. Buka konsol Amazon EC2 Global View di https://console.aws.amazon.com/ec2globalview/home.
2. Pada tab Penjelajah wilayah, di bawah Ring kasan, periksa jumlah sumber daya untuk Volume, yang mencakup jumlah volume dan jumlah Wilayah. Pilih teks yang digarisbawahi untuk melihat bagaimana jumlah volume tersebar di seluruh Wilayah.
3. Pada tab Pencarian global, pilih filter klien Jenis sumber daya = Volume. Anda dapat memfilter hasil lebih lanjut dengan menentukan Region atau tag.

AWS CLI

Untuk melihat informasi tentang volume EBS

Gunakan perintah describe-volumes. Contoh berikut menghitung volume di Wilayah saat ini.

```
aws ec2 describe-volumes --query "length(Volumes[*])"
```

Contoh berikut mencantumkan volume yang dilampirkan ke instance yang ditentukan.

```
aws ec2 describe-volumes \
    --filters "Name=attachment.instance-id,Values=i-1234567890abcdef0" \
    --query Volumes[*].VolumeId \
    --output text
```

Contoh berikut menjelaskan volume yang ditentukan.

```
aws ec2 describe-volumes --volume-ids vol-01234567890abcdef
```

Berikut ini adalah output contoh.

```
{
    "Volumes": [
        {
            "Iops": 3000,
            "VolumeType": "gp3",
            "MultiAttachEnabled": false,
            "Throughput": 125,
            "Operator": {
                "Managed": false
            },
            "VolumeId": "vol-01234567890abcdef",
            "Size": 8,
            "SnapshotId": "snap-0abcdef1234567890",
            "AvailabilityZone": "us-west-2b",
            "State": "in-use",
            "CreateTime": "2024-05-17T23:23:00.400000+00:00",
            "Attachments": [
                {
                    "DeleteOnTermination": true,
                    "VolumeId": "vol-01234567890abcdef",
                    "InstanceId": "i-1234567890abcdef0",
                    "Device": "/dev/xvda",
                    "State": "attached",
                    "AttachTime": "2024-05-17T23:23:00+00:00"
                }
            ],
            "Encrypted": false
        }
    ]
}
```

PowerShell

Untuk melihat informasi tentang volume EBS

Gunakan Get-EC2Volume cmdlet. Contoh berikut menghitung volume di Wilayah saat ini.

```
(Get-EC2Volume).Count
```

Contoh berikut mencantumkan volume yang dilampirkan ke instance yang ditentukan.

```
(Get-EC2Volume `
```

```
    -Filters @{Name="attachment.instance-id";Values="i-1234567890abcdef0"}).VolumeId
```

Contoh berikut menjelaskan volume yang ditentukan.

```
Get-EC2Volume -VolumeId vol-01234567890abcdef
```

Berikut ini adalah output contoh.

```
Attachments        : {i-1234567890abcdef0}
AvailabilityZone   : us-west-2b
CreateTime         : 5/17/2024 11:23:00 PM
Encrypted          : False
FastRestored       : False
Iops               : 3000
KmsKeyId           :
MultiAttachEnabled : False
Operator           : Amazon.EC2.Model.OperatorResponse
OutpostArn         :
Size               : 8
SnapshotId         : snap-0abcdef1234567890
SseType            :
State              : in-use
Tags               : {}
Throughput         : 125
VolumeId           : vol-01234567890abcdef
VolumeType         : gp3
```

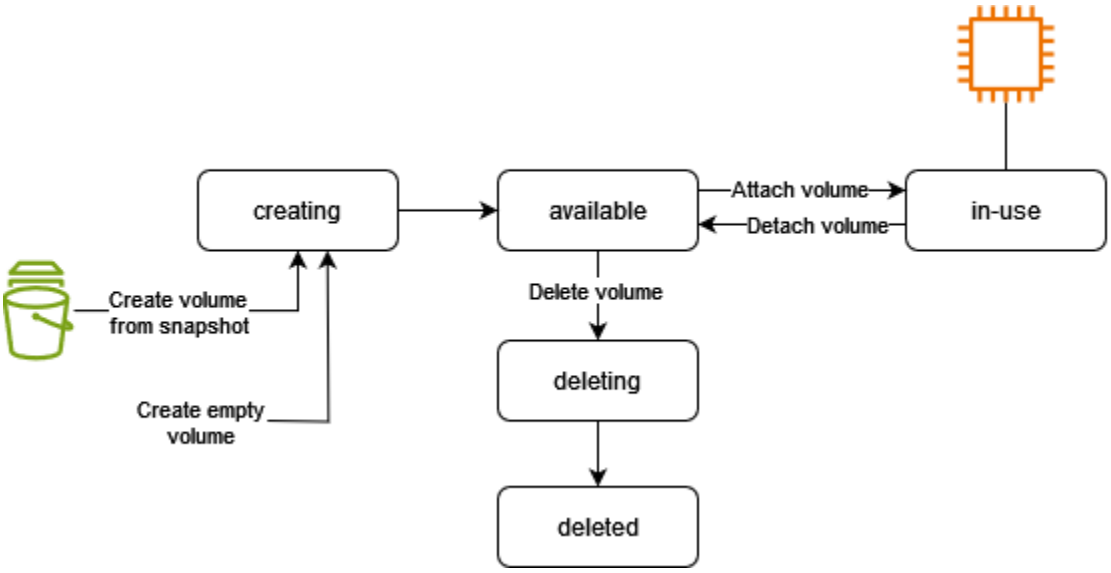
Status volume

Status volume menjelaskan ketersediaan volume Amazon EBS. Anda dapat melihat status volume di kolom Status pada halaman Volume di konsol, atau dengan menggunakan perintah describe-volume AWS CLI .

Transisi volume Amazon EBS melalui status yang berbeda dari saat dibuat hingga dihapus.

Ilustrasi berikut menunjukkan transisi antara keadaan volume. Anda dapat membuat volume dari snapshot Amazon EBS atau membuat volume kosong. Saat Anda membuat volume, itu memasuki `creating` status. Setelah volume siap digunakan, ia memasuki `available` keadaan. Anda dapat melampirkan volume yang tersedia ke instans di Zona Ketersediaan yang sama dengan

volume. Anda harus melepaskan volume sebelum Anda dapat melampirkannya ke instance lain atau menghapusnya. Anda dapat menghapus volume saat Anda tidak lagi membutuhkannya.



Tabel berikut merangkum status volume.

Status	Deskripsi
`creating`	Volume sedang dibuat.
`available`	Volume tidak terlampir pada suatu instans.
`in-use`	Volume terlampir pada suatu instans.
`deleting`	Volume sedang dihapus.
`deleted`	Volume dihapus.
`error`	Perangkat keras dasar yang terkait dengan volume EBS Anda gagal, dan data yang terkait dengan volume tidak dapat dipulihkan. Untuk informasi tentang cara mengembalikan volume atau memulihkan data pada volume, lihat Mengapa volume EBS saya memiliki status “error”? .

Lihat metrik volume

Anda bisa mendapatkan informasi tambahan tentang volume EBS Anda dari Amazon CloudWatch. Untuk informasi selengkapnya, lihat CloudWatch Metrik Amazon untuk Amazon EBS.

Lihat ruang disk kosong

Anda dapat memperoleh informasi tambahan tentang volume EBS Anda, seperti berapa banyak ruang disk yang tersedia, dari sistem operasi pada instans.

Instans Linux

Gunakan df -hT perintah dan tentukan nama perangkat:

```
[ec2-user ~]$ df -hT /dev/xvda1
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/xvda1     xfs       8.0G  1.2G  6.9G  15% /
```

Instans Windows

Anda dapat melihat ruang disk kosong dengan membuka File Explorer dan memilih PC ini.

Anda juga dapat melihat ruang kosong disk menggunakan perintah `dir` dan memeriksa baris terakhir dari output:

```
C:\> dir C:
 Volume in drive C has no label.
 Volume Serial Number is 68C3-8081

 Directory of C:\

03/25/2018  02:10 AM    <DIR>          .
03/25/2018  02:10 AM    <DIR>          ..
03/25/2018  03:47 AM    <DIR>          Contacts
03/25/2018  03:47 AM    <DIR>          Desktop
03/25/2018  03:47 AM    <DIR>          Documents
03/25/2018  03:47 AM    <DIR>          Downloads
03/25/2018  03:47 AM    <DIR>          Favorites
03/25/2018  03:47 AM    <DIR>          Links
03/25/2018  03:47 AM    <DIR>          Music
03/25/2018  03:47 AM    <DIR>          Pictures
03/25/2018  03:47 AM    <DIR>          Saved Games
```

```
03/25/2018  03:47 AM    <DIR>          Searches
03/25/2018  03:47 AM    <DIR>          Videos
               0 File(s)              0 bytes
              13 Dir(s)  18,113,662,976 bytes free
```

Anda juga dapat melihat ruang kosong disk menggunakan perintah `fsutil` berikut:

```
C:\> fsutil volume diskfree C:
Total # of free bytes        : 18113204224
Total # of bytes             : 32210153472
Total # of avail free bytes  : 18113204224
```

> **Tip**
>
> Anda juga dapat menggunakan CloudWatch agen untuk mengumpulkan metrik penggunaan ruang disk dari instans Amazon EC2 tanpa menghubungkan ke instans. Untuk informasi selengkapnya, lihat Membuat file konfigurasi CloudWatch agen dan Meng instal CloudWatch agen di Panduan CloudWatch Pengguna Amazon. Jika Anda perlu memantau penggunaan ruang disk untuk beberapa instans, Anda dapat menginstal dan mengonfigurasi CloudWatch agen pada instans tersebut menggunakan Manajer Sistem. Untuk informasi selengkapnya, lihat Menginstal CloudWatch agen menggunakan Manajer Sistem.

Memodifikasi volume Amazon EBS menggunakan operasi Volume Elastic

Dengan Volume Elastis Amazon EBS, Anda dapat meningkatkan ukuran volume, mengubah tipe volume, atau menyesuaikan performa volume EBS Anda. Jika instans Anda mendukung Volume Elastis, Anda dapat melakukannya tanpa melepas volume atau memulai ulang instans tersebut. Hal ini memungkinkan Anda untuk terus menggunakan aplikasi Anda saat perubahan berlaku.

Tidak ada biaya untuk mengubah konfigurasi volume. Anda dikenakan biaya untuk konfigurasi volume baru setelah modifikasi volume dimulai. Untuk informasi selengkapnya, lihat halaman Harga Amazon EBS.

Daftar Isi

- Pertimbangan-pertimbangan
- Batasan
- Persyaratan untuk modifikasi volume Amazon EBS

- Minta modifikasi volume Amazon EBS
- Pantau kemajuan modifikasi volume Amazon EBS
- Perluas sistem file setelah mengubah ukuran volume Amazon EBS

Pertimbangan-pertimbangan

- Setelah memulai modifikasi volume, Anda harus menunggu modifikasi tersebut mencapai `completed` status sebelum Anda dapat memulai modifikasi lain untuk volume yang sama. Anda dapat memodifikasi volume hingga empat kali dalam periode 24 jam bergulir, asalkan volume berada dalam `in-use` `available` status atau, dan semua modifikasi sebelumnya untuk volume tersebut`completed`. Jika Anda melebihi batas ini, Anda mendapatkan pesan kesalahan yang menunjukkan kapan Anda dapat melakukan modifikasi berikutnya.
- Modifikasi volume dilakukan berdasarkan upaya terbaik, dan dapat memakan waktu dari beberapa menit hingga beberapa jam untuk menyelesaikannya, tergantung pada konfigurasi volume yang diminta. Biasanya, volume 1-TiB dapat memakan waktu hingga enam jam untuk dimodifikasi. Namun, waktu tidak selalu berskala secara linier dengan ukuran volume - volume yang lebih besar mungkin membutuhkan waktu lebih sedikit, dan volume yang lebih kecil mungkin membutuhkan lebih banyak waktu.
- Peningkatan ukuran mulai berlaku setelah modifikasi volume mencapai `optimizing` keadaan, yang biasanya memakan waktu beberapa detik.
- Waktu modifikasi bertambah untuk volume yang tidak diinisialisasi sepenuhnya. Untuk informasi selengkapnya, lihat Secara manual menginisialisasi volume setelah pembuatan.
- Jika Anda mengubah jenis volume dari `gp2` ke `gp3` tanpa menentukan IOPS atau kinerja throughput, Amazon EBS secara otomatis menyediakan kinerja. Ini menggunakan kinerja yang setara dari `gp2` volume sumber atau `gp3` kinerja dasar, mana yang lebih tinggi.

  Misalnya, jika Anda memodifikasi `gp2` volume 500 GiB dengan 250 MiB/s throughput dan 1500 IOPS menjadi `gp3` tanpa menentukan IOPS atau kinerja throughput, Amazon EBS secara otomatis menyediakan `gp3` volume dengan 3000 IOPS (IOPS dasar) dan 250 MiB/s (untuk `gp3` mencocokkan throughput volume sumber). `gp2`
- Jika Anda menemukan pesan kesalahan saat mencoba mengubah volume EBS, atau jika Anda memodifikasi volume EBS yang dilampirkan ke tipe instans generasi sebelumnya, lakukan salah satu langkah berikut:
  - Untuk volume non-root, lepaskan volume dari instans, terapkan modifikasi, kemudian lampirkan kembali volume.

- Untuk volume root, hentikan instans, terapkan modifikasi, lalu mulai ulang instans.

Batasan

- Anda tidak dapat membatalkan permintaan modifikasi volume setelah dikirimkan.
- Anda harus menambah ukuran volume. Anda tidak dapat mengurangi ukuran volume. Namun, Anda dapat membuat volume yang lebih kecil dan kemudian memigrasikan data Anda ke dalamnya menggunakan alat tingkat aplikasi seperti rsync (instans Linux) atau robocopy (instans Windows).
- Ada batasan untuk penyimpanan agregat maksimum yang dapat diminta di modifikasi volume. Untuk informasi selengkapnya, lihat Kuota layanan Amazon EBS di Referensi Umum Amazon Web Services.
- Ukuran volume baru tidak dapat melebihi kapasitas yang didukung dari sistem file dan skema partisi. Untuk informasi selengkapnya, lihat Batasan volume Amazon EBS.
- Jika Anda tidak mengubah tipe volume, ukuran volume dan modifikasi performa harus dalam batas tipe volume saat ini. Jika Anda mengubah jenis volume, maka ukuran volume dan modifikasi kinerja harus berada dalam batas jenis volume target. Untuk informasi selengkapnya, lihat Tipe volume Amazon EBS
- Nitro-based inst ans mendukung volume yang disediakan hingga 256.000 IOPS. Tipe instans lainnya dapat dilampirkan ke volume yang disediakan hingga 64.000 IOPS, tetapi dapat mencapai hingga 32.000 IOPS.
- Untuk volume yang Multi-Attach diaktifkan, Anda dapat mengubah jenis volume antara `io1` dan `io2` di kedua arah, termasuk saat volume terpasang. Perubahan jenis volume lainnya tidak didukung.

  Saat Anda mengubah jenis volume antara `io1` dan `io2` untuk volume yang Multi-Attach diaktifkan, status pengaktifan reservasi NVMe volume diteruskan ke jenis volume target.
- Volume root tipe `io1`, `io2`, `gp2`, `gp3`, atau `standard` tidak dapat dimodifikasi menjadi volume `st1` atau `sc1`, bahkan jika itu dilepas dari instans.
- Jika volume dipasang sebelum 3 November 2016 pukul 23:40 UTC, Anda harus menginisialisasi dukungan Volume Elastis. Untuk informasi selengkapnya, lihat Menginisialisasi dukungan Volume Elastis.
- Meskipun instans `m3.medium` sepenuhnya mendukung modifikasi volume, `m3.large`, `m3.xlarge`, dan instans `m3.2xlarge` mungkin tidak mendukung semua fitur modifikasi volume.

Persyaratan untuk modifikasi volume Amazon EBS

Persyaratan dan batasan berikut berlaku ketika Anda memodifikasi volume Amazon EBS. Untuk mempelajari selengkapnya tentang persyaratan umum untuk volume EBS, lihat Batasan volume Amazon EBS.

Topik

- Tipe instans yang didukung
- Sistem operasi

Tipe instans yang didukung

Volume Elastis mendukung pada instans berikut:

- Semua inst ans generasi saat ini
- Instans generasi sebelumnya sebagai berikut: C1, C3, C4, G2, I2, M1, M3, M4, R3, dan R4

Jika tipe instans Anda tidak mendukung Volume Elastis, lihat Modifikasi volume EBS jika Volume Elastis tidak mendukungnya.

Sistem operasi

Persyaratan sistem operasi berikut berlaku:

Linux

AMI Linux memerlukan tabel partisi GUID (GPT) dan GRUB 2 untuk volume boot 2 TiB (2.048 GiB) atau lebih besar. Banyak AMI Linux saat ini yang masih menggunakan skema partisi MBR, yang hanya mendukung ukuran volume boot hingga 2 TiB. Jika instans Anda tidak melakukan booting dengan volume boot yang lebih besar dari 2 TiB, AMI yang Anda gunakan mungkin terbatas pada ukuran volume boot kurang dari 2 TiB. Non-boot volume tidak memiliki batasan ini pada instance Linux.

Sebelum mencoba mengubah ukuran volume booting melebihi 2 TiB, Anda dapat menentukan apakah volume tersebut menggunakan partisi MBR atau GPT dengan menjalankan perintah berikut pada instans Anda:

```
[ec2-user ~]$ sudo gdisk -l /dev/xvda
```

Instans Amazon Linux dengan partisi GPT mengembalikan informasi berikut:

```
GPT fdisk (gdisk) version 0.8.10

  Partition table scan:
    MBR: protective
    BSD: not present
    APM: not present
    GPT: present

  Found valid GPT with protective MBR; using GPT.
```

Instans SUSE dengan partisi MBR mengembalikan informasi berikut:

```
GPT fdisk (gdisk) version 0.8.8

  Partition table scan:
    MBR: MBR only
    BSD: not present
    APM: not present
    GPT: not present
```

Windows

Secara default, Windows menginisialisasi volume dengan tabel partisi Master Boot Record (MBR). Karena MBR hanya mendukung volume yang lebih kecil dari 2 TiB (2.048 GiB), Windows mencegah Anda mengubah ukuran volume MBR melebihi batas ini. Dalam kasus seperti itu, opsi Perpanjang Volume dinonaktifkan di utilitas Manajemen Disk Windows. Jika Anda menggunakan Konsol Manajemen AWS atau AWS CLI untuk membuat MBR-partitioned volume yang melebihi batas ukuran, Windows tidak dapat mendeteksi atau menggunakan ruang tambahan.

Untuk mengatasi keterbatasan ini, Anda dapat membuat volume baru yang lebih besar dengan tabel partisi GUID (GPT) dan menyalin data dari volume MBR asli.

Untuk membuat volume GPT

1. Buat volume kosong baru dengan ukuran yang diinginkan di Zona ketersediaan instans EC2 dan lampirkan ke instans Anda.

> **Note**
>
> Volume baru tidak boleh berupa volume yang dipulihkan dari snapshot.

2. Masuk ke sistem Windows Anda dan buka Manajemen Disk (diskmgmt.exe).
3. Buka menu konteks (klik kanan) untuk disk baru dan pilih Online.
4. Di jendela Inisialisasi Disk, pilih disk baru dan pilih GPT (Tabel Partisi GUID), OK.
5. Saat inisialisasi selesai, salin data dari volume asli ke volume baru, menggunakan alat seperti robocopy atau teracopy.
6. Di Manajemen Disk, ubah huruf drive ke nilai yang sesuai dan ambil volume lama secara offline.
7. Di konsol Amazon EC2 lepaskan volume lama dari instans, boot ulang instans untuk memverifikasi bahwa berfungsi dengan benar, dan menghapus volume lama.

Minta modifikasi volume Amazon EBS

Dengan Elastic Volume, Anda dapat secara dinamis meningkatkan ukuran, menambah atau mengurangi kinerja, dan mengubah tipe volume dari volume Amazon EBS Anda tanpa melepaskannya.

Gambaran umum proses

1. (Opsional) Sebelum memodifikasi volume yang berisi data berharga, praktik terbaiknya adalah membuat snapshot volume jika Anda perlu membatalkan perubahan. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS.
2. Minta modifikasi volume.
3. Memantau kemajuan modifikasi volume. Untuk informasi selengkapnya, lihat Pantau kemajuan modifikasi volume Amazon EBS.
4. Jika ukuran volume dimodifikasi, perluas sistem file volume untuk memanfaatkan peningkatan kapasitas penyimpanan. Untuk informasi selengkapnya, lihat Perluas sistem file setelah mengubah ukuran volume Amazon EBS.

Daftar Isi

- Modifikasi volume EBS menggunakan Volume Elastis
- Modifikasi volume EBS jika Volume Elastis tidak mendukungnya

- Menginisialisasi dukungan Volume Elastis (jika diperlukan)

Modifikasi volume EBS menggunakan Volume Elastis

Sebelum Anda mulai, lihat yang berikut ini:

- Pertimbangan-pertimbangan
- Batasan
- Persyaratan

Console

Untuk memodifikasi volume EBS

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume yang akan dimodifikasi dan pilih Tindakan, Ubah volume.
4. Layar Ubah volume menampilkan ID volume dan konfigurasi volume saat ini volume, termasuk tipe, ukuran, IOPS, dan throughput. Atur nilai konfigurasi baru sebagai berikut:
   - Untuk mengubah tipe, pilih nilai untuk Tipe Volume.
   - Untuk memodifikasi ukuran, masukkan nilai baru untuk Ukuran.
   - (`gp3`, `io1`, dan `io2` saja) Untuk memodifikasi IOPS, masukkan nilai baru untuk IOPS.
   - (`gp3` saja) Untuk memodifikasi throughput, masukkan nilai baru untuk Throughput.
5. Setelah Anda selesai mengubah pengaturan volume, pilih Modifikasi. Saat diminta konfirmasi, pilih Ubah.
6. Jika Anda telah meningkatkan ukuran volume, Anda juga harus memperluas partisi volume untuk memanfaatkan kapasitas penyimpanan tambahan. Untuk informasi selengkapnya, lihat Perluas sistem file setelah mengubah ukuran volume Amazon EBS.
7. (Hanya instans Windows) Jika Anda menambah ukuran volume NVMe pada instance yang tidak memiliki driver AWS NVMe, Anda harus me-reboot instance untuk mengaktifkan Windows untuk melihat ukuran volume baru. Untuk informasi selengkapnya tentang menginstal driver AWS NVMe, lihat driver AWS NVMe.

AWS CLI

Untuk memodifikasi volume EBS

Gunakan perintah modify-volume. Misalnya, jika Anda memiliki volume tipe gp2 dengan ukuran 100 GiB, contoh berikut mengubah konfigurasinya menjadi volume tipe io1 dengan 10.000 IOPS dan ukuran 200 GiB.

```
aws ec2 modify-volume \
    --volume-id vol-01234567890abcdef \
    --volume-type io1 \
    --iops 10000 \
    --size 200
```

Berikut ini adalah output contoh.

```
{
    "VolumeModification": {
        "TargetSize": 200,
        "TargetVolumeType": "io1",
        "ModificationState": "modifying",
        "VolumeId": "vol-01234567890abcdef",
        "TargetIops": 10000,
        "StartTime": "2022-01-19T22:21:02.959Z",
        "Progress": 0,
        "OriginalVolumeType": "gp2",
        "OriginalIops": 300,
        "OriginalSize": 100
    }
}
```

Jika Anda telah meningkatkan ukuran volume, Anda juga harus memperluas partisi volume untuk memanfaatkan kapasitas penyimpanan tambahan. Untuk informasi selengkapnya, lihat Perluas sistem file setelah mengubah ukuran volume Amazon EBS.

PowerShell

Untuk memodifikasi volume EBS

Gunakan Edit-EC2Volume cmdlet. Misalnya, jika Anda memiliki volume tipe gp2 dengan ukuran 100 GiB, contoh berikut mengubah konfigurasinya menjadi volume tipe io1 dengan 10.000 IOPS dan ukuran 200 GiB.

```
Edit-EC2Volume `
    -VolumeId vol-01234567890abcdef `
    -VolumeType io1 `
    -Iops 10000 `
    -Size 200
```

Jika Anda telah meningkatkan ukuran volume, Anda juga harus memperluas partisi volume untuk memanfaatkan kapasitas penyimpanan tambahan. Untuk informasi selengkapnya, lihat Perluas sistem file setelah mengubah ukuran volume Amazon EBS.

Modifikasi volume EBS jika Volume Elastis tidak mendukungnya

Jika Anda menggunakan tipr instans yang didukung, Anda dapat menggunakan Volume Elastis untuk secara dinamis mengubah ukuran, performa, dan tipe volume Amazon EBS Anda tanpa melepaskannya.

Jika Anda tidak dapat menggunakan Volume Elastis, tetapi Anda perlu memodifikasi volume root (boot), Anda harus menghentikan instans, memodifikasi volume, kemudian memulai ulang instansnya.

Setelah instans dimulai, Anda dapat memeriksa ukuran sistem file untuk melihat apakah instans Anda mengenali ruang volume yang lebih besar. Di Linux, gunakan df -h perintah untuk memeriksa ukuran sistem file.

```
[ec2-user ~]$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/xvda1            7.9G  943M  6.9G  12% /
tmpfs                 1.9G     0  1.9G   0% /dev/shm
```

Jika ukuran tidak mencerminkan volume yang baru diperluas, Anda harus memperluas sistem file perangkat Anda sehingga instans Anda dapat menggunakan ruang baru. Untuk informasi selengkapnya, lihat Perluas sistem file setelah mengubah ukuran volume Amazon EBS.

Dengan instans Windows, Anda mungkin harus membawa volume secara online untuk menggunakannya. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan. Anda tidak perlu memformat ulang volume.

Menginisialisasi dukungan Volume Elastis (jika diperlukan)

Sebelum Anda dapat memodifikasi volume yang telah dipasang ke suatu instans sebelum 3 November 2016 pada 23:40 UTC, Anda harus inisialisasi dukungan modifikasi volume menggunakan salah satu tindakan berikut:

- Lepaskan dan pasang volume
- Hentikan dan mulai instans

Console

Untuk menentukan apakah instance Anda siap

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih ikon Show/Hide Kolom (roda gigi). Pilih Waktu peluncuran dan kemudian pilih Konfirmasi.
4. Sortir daftar instans berdasarkan Waktu Peluncuran yang berbeda. Untuk setiap instans yang dimulai sebelum tanggal batas, pilih Penyimpanan dan periksa Waktu lampiran untuk melihat kapan volume dilampirkan.

AWS CLI

Untuk menentukan apakah instance Anda siap

Gunakan perintah describe-instances berikut ini untuk menentukan apakah volume dipasang sebelum 3 November 2016 23:40 UTC.

```
aws ec2 describe-instances \
    --query "Reservations[*].Instances[*].
[InstanceId,LaunchTime<='2016-11-01',BlockDeviceMappings[*]
[Ebs.AttachTime<='2016-11-01']]" \
    --output text
```

Baris pertama output untuk setiap instans menampilkan ID dan apakah itu dimulai sebelum batas tanggal (Benar atau Salah). Baris pertama diikuti dengan satu baris atau lebih yang menunjukkan jika setiap volume EBS dipasang sebelum tanggal batas (Benar atau Salah). Dalam contoh output berikut, Anda harus menginisialisasi modifikasi volume untuk instans pertama karena itu dimulai

sebelum tanggal batas dan volume root dipasang sebelum tanggal batas. Instans lainnya sudah siap karena sudah dimulai setelah tanggal batas.

```
i-e905622e              True
True
i-719f99a8              False
True
i-006b02c1b78381e57     False
False
False
i-e3d172ed              False
True
```

PowerShell

Untuk menentukan apakah sebuah instance sudah siap

Gunakan Get-EC2Instance cmdlet untuk menentukan apakah volume dilampirkan sebelum tanggal 3 November 2016 23:40 UTC.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.BlockDeviceMappings | `
     Format-Table @{Name="VolumeId";Expression={$_.Ebs.VolumeId}}, `
                  @{Name="AttachTime";Expression={$_.Ebs.AttachTime}}
```

Berikut ini adalah output contoh.

```
VolumeId              AttachTime
--------              ----------
vol-0b243c8d927752d2b 3/23/2020 12:21:14 AM
vol-043eadbeb4a8387c3 9/5/2020 7:39:22 PM
vol-0c3f0c4e55c082753 4/23/2019 4:07:40 PM
```

Pantau kemajuan modifikasi volume Amazon EBS

Saat Anda memodifikasi volume EBS, perubahan itu akan melewati urutan status. Volume memasuki status `modifying`, status `optimizing`, dan terakhir status `completed`. Pada titik ini, volume siap untuk dimodifikasi selengkapnya.

Saat volume berada pada status `optimizing`, kinerja volume Anda ada di antara spesifikasi konfigurasi sumber dan target. Performa volume transisi tidak akan kurang dari performa volume

sumber. Jika Anda menurunkan IOPS, performa volume transisi tidak kurang dari performa volume target.

Perubahan modifikasi volume berlaku sebagai berikut:

- Peningkatan ukuran mulai berlaku setelah modifikasi volume mencapai `optimizing` keadaan, yang biasanya memakan waktu beberapa detik.
- Perubahan kinerja (IOPS dan throughput) dapat memakan waktu dari beberapa menit hingga beberapa jam untuk diselesaikan, tergantung pada konfigurasi volume yang diminta. Biasanya, volume 1-TiB yang digunakan sepenuhnya dapat memakan waktu sekitar 6 jam untuk bermigrasi ke konfigurasi kinerja baru. Dalam beberapa kasus, diperlukan waktu lebih dari 24 jam agar konfigurasi kinerja baru mulai berlaku, seperti ketika volume belum sepenuhnya diinisialisasi.

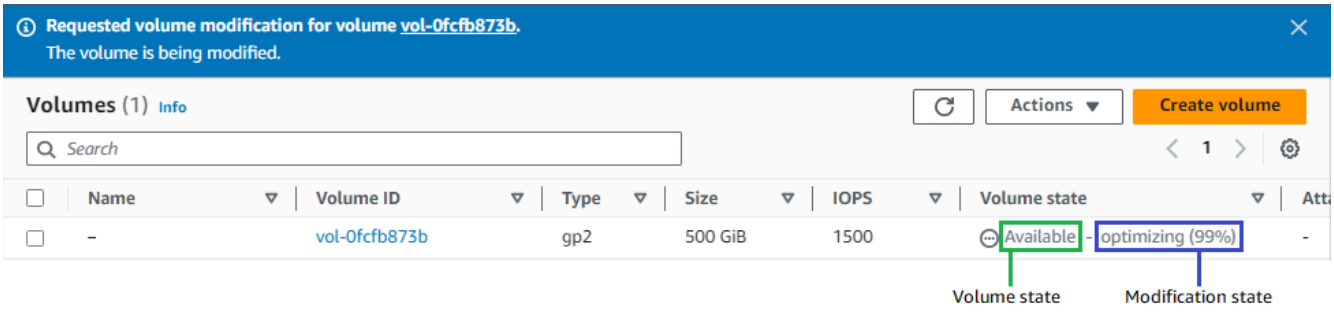
Status volume yang mungkin adalah `creating`, `available`, `in-use`, `deleting`, `deleted`, dan `error`.

Status modifikasi yang mungkin adalah `modifying`, `optimizing`, dan `completed`.

Console

Untuk memantau kemajuan modifikasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume.
4. Kolom status volume dan bidang status volume di tab Detail berisi informasi dalam format berikut: *`Volume state`* - *`Modification state`* (*`Modification progress`*%). Gambar berikut menunjukkan status modifikasi volume dan volume.



Setelah modifikasi selesai, hanya status volume yang ditampilkan. Status modifikasi dan kemajuan tidak lagi ditampilkan.

Atau, Anda dapat menggunakan Amazon EventBridge untuk membuat aturan notifikasi untuk peristiwa modifikasi volume. Untuk informasi selengkapnya, lihat Mem ulai dengan Amazon EventBridge.

AWS CLI

Untuk memantau kemajuan modifikasi

Gunakan describe-volumes-modifications untuk melihat kemajuan dari satu atau lebih modifikasi volume. Contoh berikut menjelaskan modifikasi volume untuk dua volume.

```
aws ec2 describe-volumes-modifications \
    --volume-ids vol-1111111111111111 vol-2222222222222222
```

Dalam contoh output berikut, modifikasi volume masih dalam status `modifying`. Kemajuan dilaporkan dalam bentuk persentase.

```
{
    "VolumesModifications": [
        {
            "TargetSize": 200,
            "TargetVolumeType": "io1",
            "ModificationState": "modifying",
            "VolumeId": "vol-1111111111111111",
            "TargetIops": 10000,
            "StartTime": "2017-01-19T22:21:02.959Z",
            "Progress": 0,
            "OriginalVolumeType": "gp2",
            "OriginalIops": 300,
            "OriginalSize": 100
        },
        {
            "TargetSize": 2000,
            "TargetVolumeType": "sc1",
            "ModificationState": "modifying",
            "VolumeId": "vol-2222222222222222",
            "StartTime": "2017-01-19T22:23:22.158Z",
            "Progress": 0,
            "OriginalVolumeType": "gp2",
            "OriginalIops": 300,
            "OriginalSize": 1000
```

```
        }
    ]
}
```

Contoh berikutnya menggambarkan semua volume dengan status modifikasi `optimizing` atau `completed`, lalu memfilter dan memformat hasil untuk hanya menampilkan modifikasi yang dimulai pada atau setelah 1 Februari 2017:

```
aws ec2 describe-volumes-modifications \
    --filters Name=modification-state,Values="optimizing","completed" \
    --query "VolumesModifications[?StartTime>='2017-02-01'].
{ID:VolumeId,STATE:ModificationState}"
```

Berikut ini adalah contoh output dengan informasi tentang dua volume:

```
[
    {
        "STATE": "optimizing",
        "ID": "vol-06397e7a0eEXAMPLE"
    },
    {
        "STATE": "completed",
        "ID": "vol-ba74e18c2aEXAMPLE"
    }
]
```

PowerShell

Untuk memantau kemajuan modifikasi

Gunakan Get-EC2VolumeModification cmdlet. Contoh berikut menjelaskan modifikasi volume untuk dua volume.

```
Get-EC2VolumeModification `
    -VolumeId vol-1111111111111111 vol-2222222222222222
```

> **Note**
>
> Jarang, AWS kesalahan sementara dapat mengakibatkan `failed` keadaan. Ini bukan indikasi kesehatan volume; itu hanya menunjukkan bahwa modifikasi terhadap volume gagal. Jika ini terjadi, coba kembali modifikasi volume.

Perluas sistem file setelah mengubah ukuran volume Amazon EBS

Setelah Anda meningkatkan ukuran volume EBS, Anda harus memperluas partisi dan sistem file ke ukuran baru yang lebih besar. Anda dapat melakukan ini segera setelah volume memasuki status `optimizing`.

Sebelum Anda mulai

- Buat snapshot dari volume, jika Anda perlu mengembalikan perubahan Anda. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS.
- Konfirmasikan bahwa modifikasi volume berhasil dan berada dalam status `optimizing` atau `completed`. Untuk informasi selengkapnya, lihat Pantau kemajuan modifikasi volume Amazon EBS.
- Pastikan volume terpasang ke instance dan diformat dan dipasang. Untuk informasi selengkapnya, lihat Format dan pasang volume yang terpasang.
- (Hanya instans Linux) Jika Anda menggunakan volume logis pada volume Amazon EBS, Anda harus menggunakan Logical Volume Manager (LVM) untuk memperluas volume logis. Untuk petunjuk tentang cara melakukan ini, lihat bagian Per panjang LV di artikel Bagaimana cara menggunakan LVM untuk membuat volume logis pada partisi volume EBS? .

Instans Linux

> **Note**
>
> Instruksi berikut memandu Anda melalui proses memperluas sistem file XFS dan Ext4 untuk Linux. Untuk informasi tentang memperluas sistem file yang berbeda, lihat dokumentasinya.

Sebelum Anda dapat memperluas sistem file di Linux, Anda harus memperpanjang partisi, jika volume Anda memilikinya.

Perluas sistem file volume EBS

Gunakan prosedur berikut untuk memperluas sistem file untuk volume yang diubah ukurannya.

Perhatikan bahwa penamaan perangkat dan partisi berbeda untuk instans Xen dan instans yang dibangun di Nitro System. Untuk menentukan apakah instans Anda adalah Xen-based atau Nitro-based, lihat jenis hypervisor Amazon EC2.

Untuk memperluas sistem file volume EBS

1. Terhubung ke instans Anda.
2. Ubah ukuran partisi, jika diperlukan. Untuk melakukannya:

   a. Periksa apakah volume memiliki partisi. Gunakan perintah lsblk.

      Nitro instance example

      Dalam contoh output berikut, volume root (nvme0n1) memiliki dua partisi (nvme0n1p1 dan nvme0n1p128), sedangkan volume tambahan (nvme1n1) tidak memiliki partisi.

```
[ec2-user ~]$ sudo lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme1n1       259:0    0  30G  0 disk /data
nvme0n1       259:1    0  16G  0 disk
##nvme0n1p1   259:2    0   8G  0 part /
##nvme0n1p128 259:3    0   1M  0 part
```

      Xen instance example

      Dalam contoh output berikut, volume root (xvda) memiliki satu partisi (xvda1), sedangkan volume tambahan (xvdf) tidak memiliki partisi.

```
[ec2-user ~]$ sudo lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  16G  0 disk
##xvda1 202:1    0   8G  0 part /
xvdf    202:80   0  24G  0 disk
```

- Jika volume memiliki partisi, lanjutkan ke langkah berikutnya (2b).
- Jika volume tidak memiliki partisi, lewati langkah 2b, 2c, dan 2d, dan lanjutkan ke langkah 3.

> **Tip pemecahan masalah**
>
> Jika Anda tidak melihat volume dalam output perintah, pastikan bahwa volume terpasang ke instance, dan bahwa itu diformat dan dipasang.

b. Periksa apakah partisi perlu diperpanjang. Pada output perintah lsblk dari langkah sebelumnya, bandingkan ukuran partisi dan ukuran volume.

- Jika ukuran partisi lebih kecil dari ukuran volume, lanjutkan ke langkah berikutnya (2c).
- Jika ukuran partisi sama dengan ukuran volume, partisi tidak perlu diperpanjang - lewati langkah 2c dan 2d, dan lanjutkan ke langkah 3.

> **Tip pemecahan masalah**
>
> Jika volume masih mencerminkan ukuran aslinya, konfirmasikan bahwa modifikasi volume berhasil.

c. Perluas partisi. Gunakan growpart perintah dan tentukan nama perangkat dan nomor partisi.

Nitro instance example

Nomor partisi adalah nomor setelahp. Misalnya, untuk`nvme0n1p1`, nomor partisi adalah`1`. Untuk`nvme0n1p128`, nomor partisi adalah`128`.

Untuk memperluas partisi bernama`nvme0n1p1`, gunakan perintah berikut.

> **Important**
>
> Perhatikan ruang antara nama perangkat (`nvme0n1`) dan nomor partisi (`1`).

```
[ec2-user ~]$ sudo growpart /dev/nvme0n1 1
```

Xen instance example

Nomor partisi adalah nomor setelah nama perangkat. Misalnya, untuk`xvda1`, nomor partisi adalah`1`. Untuk`xvda128`, nomor partisi adalah`128`.

Untuk memperluas partisi bernama`xvda1`, gunakan perintah berikut.

> **Important**
>
> Perhatikan ruang antara nama perangkat (`xvda`) dan nomor partisi (`1`).

```
[ec2-user ~]$ sudo growpart /dev/xvda 1
```

> **Tip pemecahan masalah**
>
> - `mkdir: cannot create directory ‘/tmp/growpart.31171’: No space left on device FAILED: failed to make temp dir`: Menunjukkan bahwa tidak ada cukup ruang disk kosong pada volume bagi growpart untuk membuat direktori sementara yang diperlukan untuk melakukan pengubahan ukuran. Kosongkan ruang disk, kemudian coba lagi.
> - `must supply partition-number`: Menunjukkan bahwa Anda menentukan partisi yang salah. Gunakan lsblk perintah untuk mengonfirmasi nama partisi, dan pastikan Anda memasukkan spasi antara nama perangkat dan nomor partisi.
> - `NOCHANGE: partition 1 is size 16773087. it cannot be grown`: Menunjukkan bahwa partisi sudah memperluas seluruh volume dan tidak dapat diperpanjang. Konfirmasikan bahwa modifikasi volume berhasil.

d. Verifikasi bahwa partisi telah diperpanjang. Gunakan perintah lsblk. Ukuran partisi sekarang harus sama dengan ukuran volume.

Nitro instance example

Contoh output berikut menunjukkan bahwa volume (`nvme0n1`) dan partisi (`nvme0n1p1`) berukuran sama (`16 GB`).

```
[ec2-user ~]$ sudo lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme1n1       259:0    0  30G  0 disk /data
nvme0n1       259:1    0  16G  0 disk
##nvme0n1p1   259:2    0  16G  0 part /
##nvme0n1p128 259:3    0   1M  0 part
```

Xen instance example

Contoh output berikut menunjukkan bahwa volume (`xvda`) dan partisi (`xvda1`) berukuran sama (`16 GB`).

```
[ec2-user ~]$ sudo lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  16G  0 disk
##xvda1 202:1    0  16G  0 part /
xvdf    202:80   0  24G  0 disk
```

3. Perluas sistem file.

   a. Dapatkan nama, ukuran, tipe, dan titik pemasangan untuk sistem file yang perlu Anda perluas. Gunakan lsblk -f perintah df -hT atau.

   Nitro instance example

   Contoh output berikut untuk df -hT perintah menunjukkan bahwa sistem `/dev/nvme0n1p1` file berukuran 8 GB, jenisnya`xfs`, dan titik pemasangan adalah`/`.

```
[ec2-user ~]$ df -hT
Filesystem      Type  Size  Used Avail Use% Mounted on
/dev/nvme0n1p1  xfs   8.0G  1.6G  6.5G  20% /
/dev/nvme1n1    xfs   8.0G   33M  8.0G   1% /data
...
```

Xen instance example

Contoh output berikut untuk df -hT perintah menunjukkan bahwa sistem `/dev/xvda1` file berukuran 8 GB, jenisnya`ext4`, dan titik pemasangan adalah`/`.

```
[ec2-user ~]$ df -hT
Filesystem      Type   Size    Used   Avail   Use%   Mounted on
/dev/xvda1      ext4   8.0G    1.9G   6.2G    24%    /
/dev/xvdf1      xfs    24.0G   45M    8.0G    1%     /data
...
```

- Jika ukuran sistem file lebih kecil dari ukuran volume, lanjutkan ke langkah berikutnya (3b).
- Jika ukuran sistem file sama dengan ukuran volume, maka tidak perlu diperpanjang. Dalam hal ini, lewati langkah-langkah yang tersisa - partisi dan sistem file telah diperluas ke ukuran volume baru.

b. Perintah untuk memperluas sistem file berbeda bergantung pada jenis sistem file. Pilih perintah yang benar berikut berdasarkan tipe sistem file yang Anda ketahui di langkah sebelumnya.

- [Sistem file XFS] Gunakan perintah xfs_growfs dan tentukan titik pemasangan sistem file yang Anda ketahui pada langkah sebelumnya.

  Nitro and Xen instance example

  Misalnya, untuk memperluas sistem file yang dipasang pada `/`, gunakan perintah berikut.

  ```
  [ec2-user ~]$ sudo xfs_growfs -d /
  ```

  Tip pemecahan masalah

  - `xfs_growfs: /data is not a mounted XFS filesystem`: Menunjukkan bahwa Anda menentukan titik pemasangan yang salah, atau sistem file bukan XFS. Untuk memverifikasi titik pemasangan dan tipe sistem file, gunakan perintah df -hT.

- `data size unchanged, skipping`: Menunjukkan bahwa sistem file sudah memperluas seluruh volume. Jika volume tidak memiliki partisi, konfirmasikan bahwa modifikasi volume berhasil. Jika volume memiliki partisi, pastikan partisi diperpanjang seperti yang dijelaskan pada langkah 2.

- [Sistem file Ext4] Gunakan perintah resize2fs dan tentukan nama sistem file yang Anda ketahui pada langkah sebelumnya.

  Nitro instance example

  Misalnya, untuk memperluas sistem file yang dipasang yang bernama `/dev/nvme0n1p1`, gunakan perintah berikut.

  ```
  [ec2-user ~]$ sudo resize2fs /dev/nvme0n1p1
  ```

  Xen instance example

  Misalnya, untuk memperluas sistem file yang dipasang yang bernama `/dev/xvda1`, gunakan perintah berikut.

  ```
  [ec2-user ~]$ sudo resize2fs /dev/xvda1
  ```

  Tip pemecahan masalah

  - `resize2fs: Bad magic number in super-block while trying to open /dev/xvda1`: Menunjukkan bahwa sistem file bukan Ext4. Untuk memverifikasi tipe sistem file, gunakan perintah df -hT.
  - `open: No such file or directory while opening /dev/xvdb1`: Menunjukkan bahwa Anda menentukan partisi yang salah. Untuk memverifikasi partisi, gunakan perintah df -hT.
  - `The filesystem is already 3932160 blocks long. Nothing to do!`: Menunjukkan bahwa sistem file sudah memperluas seluruh volume. Jika volume tidak memiliki partisi, konfirmasikan bahwa modifikasi volume berhasil. Jika volume memiliki partisi, pastikan partisi diperpanjang, seperti yang dijelaskan pada langkah 2.

- [Sistem file lainnya] Lihat dokumentasi untuk sistem file Anda untuk mendapatkan petunjuk.

c. Verifikasi bahwa sistem file telah diperluas. Gunakan perintah df -hT dan konfirmasikan bahwa ukuran sistem file sama dengan ukuran volume.

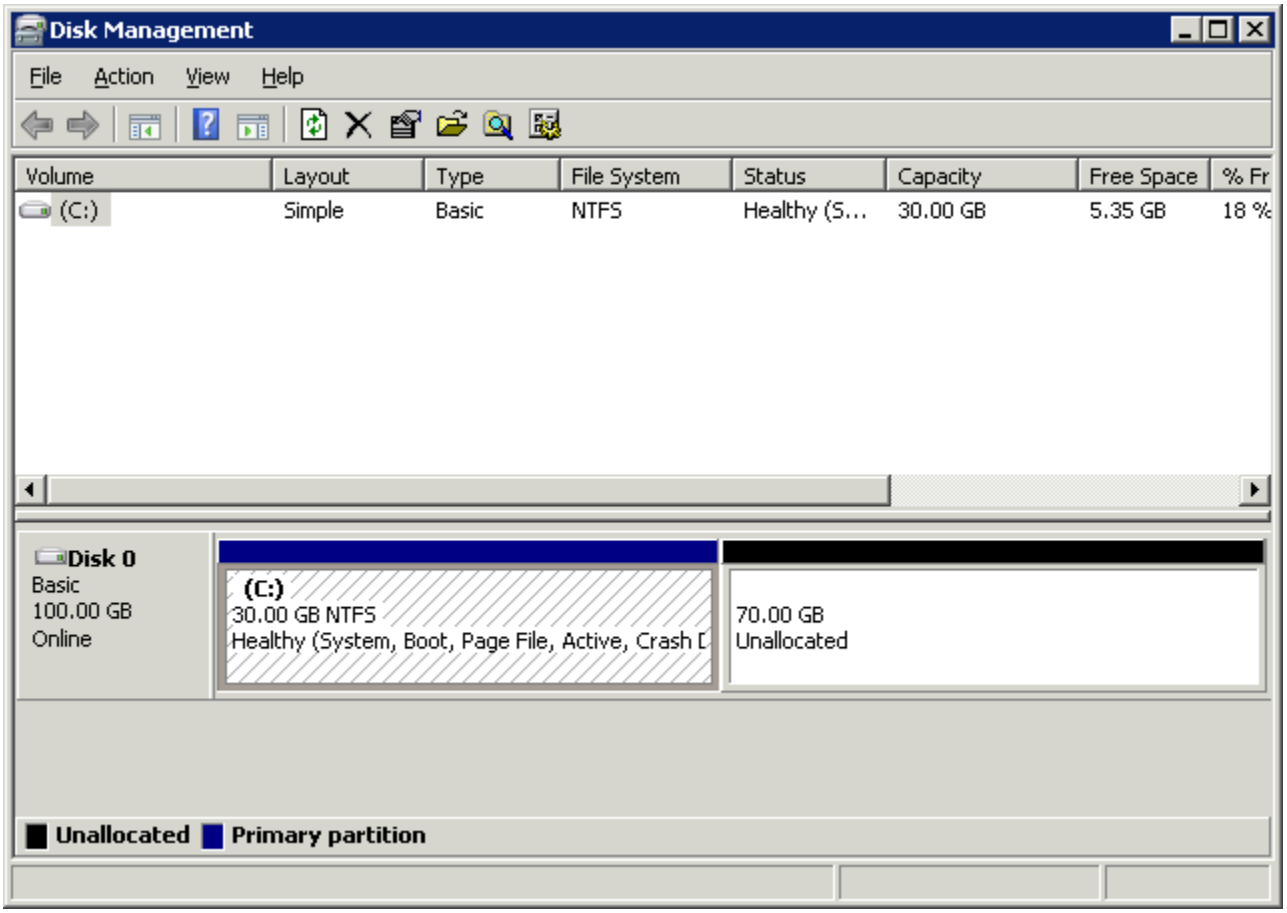
Instans Windows

Gunakan salah satu metode berikut untuk memperluas sistem file pada instance Windows.

Disk Management utility

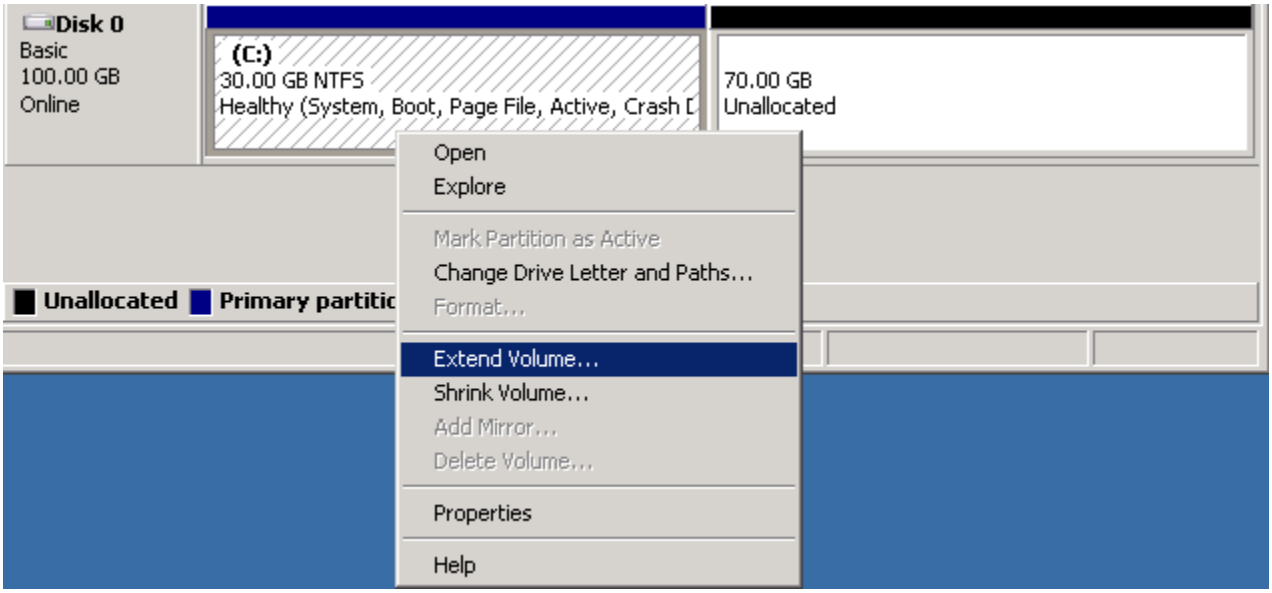
Untuk memperluas sistem file menggunakan Manajemen Disk

1. Sebelum memperluas sistem file yang berisi data berharga, praktik terbaiknya adalah membuat snapshot volume yang berisi data tersebut jika Anda perlu mengembalikan perubahan Anda. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS.
2. Masuk ke instans Windows menggunakan Remote Desktop.
3. Pada dialog Jalankan, masukkan diskmgmt.msc dan tekan Enter. Utilitas Manajemen Disk terbuka.

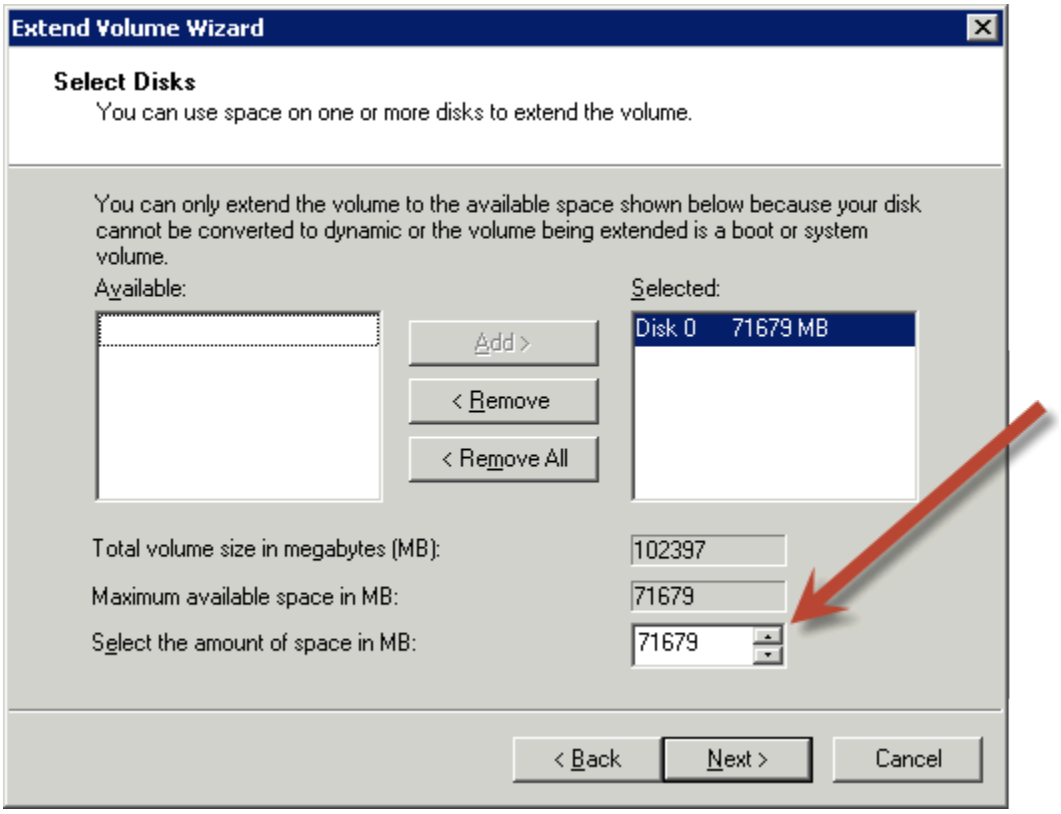


4. Di menu Manajemen Disk, pilih Tindakan, Pindai Ulang Disk.
5. Buka menu konteks (klik kanan) untuk drive yang diperluas dan pilih Perluas Volume.

> **Note**
>
> Perluas volume mungkin dinonaktifkan (berwarna abu-abu) jika:
>
> - Ruang yang tidak terisi tidak berdekatan dengan drive. Ruang yang tidak terisi harus berdekatan dengan sisi kanan drive yang ingin Anda perluas.
> - Volume menggunakan gaya partisi Master Boot Record (MBR) dan ukurannya sudah 2 TB. Volume yang menggunakan MBR tidak dapat melebihi 2 TB dalam ukuran.



6. Di wizard Perluas Volume, pilih Selanjutnya. Untuk Pilih jumlah ruang dalam MB, masukkan jumlah megabyte untuk memperluas volume. Umumnya, Anda menentukan ruang maksimum yang tersedia. Teks yang disorot di bawah Dipilih adalah jumlah ruang yang ditambahkan, bukan ukuran akhir yang akan dimiliki oleh volume. Selesaikan panduan.



7. Jika Anda menambah ukuran volume NVMe pada instans yang tidak memiliki driver NVMe AWS , Anda harus melakukan boot ulang instans agar Windows dapat melihat ukuran volume yang baru. Untuk informasi selengkapnya tentang menginstal driver AWS NVMe, lihat driver AWS NVMe.

PowerShell

Gunakan prosedur berikut untuk memperluas sistem file Windows menggunakan PowerShell.

Untuk memperluas sistem file menggunakan PowerShell

1. Sebelum memperluas sistem file yang berisi data berharga, praktik terbaiknya adalah membuat snapshot volume yang berisi data tersebut jika Anda perlu mengembalikan perubahan Anda. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS.
2. Masuk ke instans Windows menggunakan Remote Desktop.
3. Jalan PowerShell kan sebagai administrator.
4. Jalankan `Get-Partition` perintah. PowerShell mengembalikan nomor partisi yang sesuai untuk setiap partisi, huruf drive, offset, ukuran, dan jenis. Perhatikan huruf drive dari partisi yang akan diperluas.

5. Jalankan perintah berikut untuk memindai ulang disk.

```
"rescan" | diskpart
```

6. Jalankan perintah berikut, menggunakan huruf drive yang Anda catat di langkah 4 sebagai pengganti**<drive-letter>**. PowerShell mengembalikan ukuran minimum dan maksimum partisi yang diizinkan, dalam byte.

```
Get-PartitionSupportedSize -DriveLetter <drive-letter>
```

7. Untuk memperpanjang partisi ke jumlah tertentu, jalankan perintah berikut, yang akan memasukkan ukuran baru volume di tempat **<size>**. Anda dapat memasukkan ukurannya `KB`, `MB`, dan `GB`; contohnya, `50GB`.

```
Resize-Partition -DriveLetter <drive-letter> -Size <size>
```

   Untuk memperpanjang partisi ke ukuran maksimum yang tersedia, jalankan perintah berikut.

```
Resize-Partition -DriveLetter <drive-letter> -Size $(Get-PartitionSupportedSize
 -DriveLetter <drive-letter>).SizeMax
```

   Per PowerShell intah berikut menunjukkan alur perintah dan respons lengkap untuk memperluas sistem file ke ukuran tertentu.

```
PS C:\> Get-Partition


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&26a12046&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                C           1048576                                     30 GB IFS


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&3476342 3&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                D           1048576                                     8 MB IFS


PS C:\> "rescan" | diskpart

Microsoft DiskPart version 10.0.17763.1911

Copyright (C) Microsoft Corporation.
On computer:

DISKPART>
Please wait while DiskPart scans your configuration...

DiskPart has finished scanning your configuration.

DISKPART>
PS C:\> Get-PartitionSupportedSize -DriveLetter D

SizeMin      SizeMax
-------      -------
8388608 107372085248


PS C:\> Resize-Partition -DriveLetter D -Size 50GB
PS C:\> Get-Partition


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&26a12046&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                C           1048576                                     30 GB IFS


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&3476342 3&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                D           1048576                                     50 GB IFS
```

Per PowerShell intah berikut menunjukkan alur perintah dan respons lengkap untuk memperluas sistem file ke ukuran maksimum yang tersedia.

```
PS C:\> Get-Partition


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&26a12046&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                C           1048576                                     30 GB IFS


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&3476423&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                D           1048576                                     50 GB IFS


PS C:\> "rescan" | diskpart

Microsoft DiskPart version 10.0.17763.1911

Copyright (C) Microsoft Corporation.
On computer:

DISKPART>
Please wait while DiskPart scans your configuration...

DiskPart has finished scanning your configuration.

DISKPART>
PS C:\> Get-PartitionSupportedSize -DriveLetter D

 SizeMin      SizeMax
 -------      -------
59047936 107372085248


PS C:\> Resize-Partition -DriveLetter D -Size $(Get-PartitionSupportedSize -DriveLetter D).SizeMax
PS C:\> Get-Partition


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&26a12046&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                C           1048576                                     30 GB IFS


   DiskPath: \\?\scsi#disk&ven_nvme&prod_amazon_elastic_b#4&3476423&0&000000#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

PartitionNumber  DriveLetter Offset                                        Size Type
---------------  ----------- ------                                        ---- ----
1                D           1048576                                    100 GB IFS
```

Melepaskan volume Amazon EBS dari instans Amazon EC2

Anda perlu melepaskan volume Amazon Elastic Block Store (Amazon EBS) dari sebuah instans sebelum Anda dapat melampirkannya ke instans yang berbeda atau menghapusnya. Menghapus volume tidak memengaruhi data pada volume.

Topik

- Pertimbangan-pertimbangan
- Lepaskan dan lepaskan volume

- Pemecahan Masalah

Pertimbangan-pertimbangan

- Anda dapat melepaskan volume Amazon EBS dari suatu instans secara eksplisit atau dengan mengakhiri instans tersebut. Namun, jika proses sedang berjalan, Anda harus melepas volume terlebih dahulu dari instans.
- Jika volume EBS adalah perangkat root suatu instans, Anda harus menghentikan instans tersebut sebelum Anda dapat melepaskan volume.
- Anda dapat melampirkan ulang volume yang Anda lepas (tanpa melepasnya), tetapi tidak akan mendapatkan titik pemasangan yang sama. Jika penulisan ke volume sedang berlangsung saat dilepas, data di volume mungkin tidak sinkron.
- Setelah Anda melepaskan volume, Anda masih dikenakan biaya untuk penyimpanan volume selama jumlah penyimpanan melebihi batas Tingkat AWS Gratis. Anda harus menghapus volume agar tidak dikenai biaya lebih lanjut. Untuk informasi selengkapnya, lihat Menghapus volume Amazon EBS.

Lepaskan dan lepaskan volume

Gunakan prosedur berikut untuk melepaskan volume dari suatu instans. Hal ini dapat berguna saat Anda perlu memasang volume ke instans yang berbeda atau saat Anda perlu menghapus volume.

Langkah-langkah

- Langkah 1: Melepaskan volume
- Langkah 2: Melepaskan volume dari instans
- Langkah 3: (Hanya instans Windows) Copot pemasangan lokasi perangkat offline

Langkah 1: Melepaskan volume

Instans Linux

Dari instans Linux Anda, gunakan perintah berikut untuk melepaskan perangkat `/dev/sdh`.

```
[ec2-user ~]$ sudo umount -d /dev/sdh
```

Instans Windows

Dari instans Windows Anda, lepaskan volume dengan cara berikut.

1. Mulai utilitas Manajemen Disk.
   - (Windows Server 2012 dan yang lebih baru) Pada bilah tugas, pilih (klik kanan) logo Windows dan pilih Manajemen Disk.
   - Windows Server 2008) Pilih Mulai, Alat Administratif, Manajemen Komputer, Manajemen Disk.
2. Pilih disk (misalnya, pilih (klik kanan) Disk 1) dan kemudian pilih Offline. Tunggu status disk untuk diubah ke Offline sebelum membuka konsol Amazon EC2.

Langkah 2: Melepaskan volume dari instans

Untuk melepaskan volume dari instans, gunakan salah satu metode berikut:

Console

Untuk melepaskan volume EBS

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume.
4. Pilih T ind akan, Lepaskan volume.
5. Ketika diminta untuk mengonfirmasi, pilih Lepaskan.

AWS CLI

Untuk melepaskan volume EBS dari instans

Setelah melepaskan volume, gunakan perintah detach-volume.

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

PowerShell

Untuk melepaskan volume EBS dari instans

Setelah melepas volume, gunakan Dismount-EC2Volume cmdlet.

```
Dismount-EC2Volume -VolumeId vol-01234567890abcdef
```

Langkah 3: (Hanya instans Windows) Copot pemasangan lokasi perangkat offline

Ketika Anda melepaskan dan melepaskan lampiran volume dari suatu instans, Windows menandai lokasi perangkat sebagai offline. Lokasi perangkat tetap offline setelah boot ulang, dan berhenti serta memulai ulang instans. Ketika Anda memulai ulang instans, Windows mungkin memasang salah satu volume yang tersisa ke lokasi perangkat offline. Hal ini menyebabkan volume tidak tersedia di Windows. Untuk mencegah hal ini terjadi dan untuk memastikan bahwa semua volume terpasang ke lokasi perangkat online saat Windows dimulai berikutnya, lakukan langkah-langkah berikut:

1. Pada instans, buka Manajer Perangkat.
2. Dalam Manajer Perangkat, pilih Lihat, Tampilkan perangkat tersembunyi.
3. Dalam daftar perangkat, perluas simpul Kontroler penyimpanan.

   Lokasi perangkat tempat volume yang dilepas lampirannya dipasang bernama `AWS NVMe Elastic Block Storage Adapter` dan akan tampak berwarna abu-abu.
4. Pilih (klik kanan) setiap lokasi perangkat berwarna abu-abu bernama`AWS NVMe Elastic Block Storage Adapter`, pilih Uninstall device dan pilih Uninstall.

   > **Important**
   >
   > Jangan pilih kotak centang Hapus perangkat lunak driver untuk perangkat ini.

Pemecahan Masalah

Berikut ini adalah masalah umum yang dihadapi saat melepaskan lampiran volume, dan cara mengatasinya.

> **Note**
>
> Untuk menjaga kemungkinan kehilangan data, ambil snapshot volume Anda sebelum mencoba melepasnya. Pelepasan paksa dari volume yang macet dapat menyebabkan kerusakan pada sistem file atau data yang ada di dalamnya atau tidak dapat melampirkan volume baru menggunakan nama perangkat yang sama, kecuali jika instans di-boot ulang.

- Jika Anda mengalami masalah saat melepaskan volume melalui konsol Amazon EC2, Anda akan terbantu dengan menggunakan perintah CLI describe-volumes untuk mendiagnosis masalah. Untuk informasi selengkapnya, lihat describe-volume.
- Jika volume Anda tetap dalam status `detaching`, Anda dapat memaksa pelepasan dengan memilih Lepas Paksa. Gunakan opsi ini hanya sebagai upaya terakhir untuk memisahkan volume dari instans yang gagal, atau jika Anda melepaskan volume dengan tujuan menghapusnya. Instans yang ada tidak memiliki peluang untuk membersihkan cache sistem file atau metadata sistem file. Jika Anda menggunakan opsi ini, Anda harus melakukan prosedur pemeriksaan dan perbaikan sistem file.
- Jika Anda telah mencoba melepas paksa volume beberapa kali selama beberapa menit tetapi tetap berada di status `detaching`, Anda dapat mengirim permintaan bantuan ke AWS re:Post. Untuk membantu mempercepat resolusi, sertakan ID volume dan jelaskan langkah-langkah yang telah Anda ambil.
- Saat Anda mencoba melepaskan volume yang masih terpasang, volume dapat macet di status `busy` saat mencoba untuk melepaskannya. Output berikut dari describe-volumes menunjukkan contoh dari kondisi ini:

```
"Volumes": [
    {
        "AvailabilityZone": "us-west-2b",
        "Attachments": [
            {
                "AttachTime": "2022-07-21T23:44:52.000Z",
                "InstanceId": "i-1234567890abcdef0",
                "VolumeId": "vol-01234567890abcdef",
                "State": "busy",
                "DeleteOnTermination": false,
                "Device": "/dev/sdf"
            }
        ...
    }
]
```

Saat Anda mengalami kondisi ini, pelepasan dapat ditunda tanpa batas waktu hingga Anda melepas volume, pelepasan paksa, boot ulang instans, atau ketiganya.

Menghapus volume Amazon EBS

Anda dapat menghapus volume Amazon EBS yang tidak lagi Anda butuhkan. Setelah penghapusan, datanya hilang dan volumenya tidak dapat dilampirkan ke instans apa pun. Jadi, sebelum dihapus Anda dapat menyimpan snapshot volume, yang dapat Anda gunakan untuk membuat ulang volume nantinya.

Anda tidak dapat menghapus volume jika terlampir ke suatu instans. Untuk menghapus volume, Anda harus melepasnya terlebih dahulu. Untuk informasi selengkapnya, lihat Melepaskan volume Amazon EBS dari instans Amazon EC2.

Jika Anda menghapus volume yang cocok dengan aturan retensi Recycle Bin, volume tersebut akan dipertahankan di Recycle Bin alih-alih segera dihapus. Untuk informasi selengkapnya, lihat Recycle Bin.

Console

Untuk menghapus volume EBS

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume. Verifikasi bahwa volume berada dalam status Ter sedia.
4. Pilih T ind akan, Hapus volume.

   Jika opsi ini dinonaktifkan, volume dilampirkan ke instance dan tidak dapat dihapus.
5. Saat diminta konfirmasi, masukkan**delete**, lalu pilih H apus.

AWS CLI

Untuk memeriksa apakah volume EBS sedang digunakan

Gunakan perintah describe-volumes. Jika volume sedang digunakan, statusnya`in-use`. Kalau tidak, memang begitu`available`.

```
aws ec2 describe-volumes \
    --volume-id vol-01234567890abcdef \
    --query Volumes[*].State \
    --output text
```

Untuk menghapus volume EBS

Gunakan perintah delete-volume.

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

PowerShell

Untuk memeriksa apakah volume EBS sedang digunakan

Gunakan Get-EC2Volume cmdlet. Jika volume sedang digunakan, statusnya`in-use`. Kalau tidak, memang begitu`available`.

```
(Get-EC2Volume `
    -VolumeId vol-01234567890abcdef).State.Value
```

Untuk menghapus volume EBS

Gunakan Remove-EC2Volume cmdlet.

```
Remove-EC2Volume -VolumeId vol-01234567890abcdef
```

Mengganti volume Amazon EBS menggunakan snapshot

Snapshot Amazon EBS adalah alat bantu pencadangan pilihan di Amazon EC2 karena kecepatan, kenyamanan, dan biayanya. Saat membuat volume dari snapshot, Anda membuat ulang statusnya pada titik waktu tertentu dengan data yang disimpan hingga titik tertentu secara utuh. Dengan memasang volume yang dibuat dari snapshot ke suatu instans, Anda dapat menduplikasi data di seluruh Wilayah, membuat lingkungan pengujian, mengganti volume produksi yang rusak atau korup secara keseluruhan, atau mengambil file dan direktori spesifik dan mentransfernya ke volume lain yang terlampir. Untuk informasi selengkapnya, lihat Snapshot Amazon EBS.

Anda dapat menggunakan salah satu prosedur berikut untuk mengganti volume Amazon EBS dengan volume lain yang dibuat dari snapshot sebelumnya dari volume tersebut.

Persyaratan

Anda harus membuat volume di Zona Ketersediaan yang sama dengan instans. Volume harus dilampirkan ke instans di Zona Ketersediaan yang sama.

Console

Untuk mengganti volume

1. Buat volume dari snapshot dan tulis ID volume baru. Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.
2. Pada halaman Instans, pilih instans untuk mengganti volume dan tuliskan ID instans.

   Dengan instans yang masih dipilih, pilih tab Penyimpanan. Di bagian Perangkat blok, cari volume yang akan diganti dan tuliskan nama perangkat untuk volume, misalnya `/dev/sda1`.
3. Pada tab Penyimpanan, pilih ID volume, lalu lepas kan dan lepaskan volume dari instance.
4. Pilih volume baru yang Anda buat pada langkah 1 dan pilih Tindakan, Pasang volume.

   Untuk Instans dan Nama perangkat, masukkan ID instans dan nama perangkat yang Anda tulis di Langkah 2, lalu pilih Pasang volume.
5. Sambungkan ke instans dan pasang volume. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

AWS CLI

Untuk mengganti volume

1. Buat snapshot volume baru dari snapshot. Gunakan perintah create-volume dengan opsi. `--snapshot-id` Untuk `--availability-zone`, tentukan Zona Ketersediaan yang sama dengan instans. Perhatikan ID volume baru dalam output.

```
aws ec2 create-volume \
    --volume-type gp3 \
    --snapshot-id snap-0abcdef1234567890 \
    --availability-zone us-east-1a
```

2. Dapatkan nama perangkat volume yang akan diganti. Gunakan perintah describe-instances. Untuk `--instance-ids`, tentukan ID instans tempat mengganti volume. Perhatikan nama perangkat dan ID volume volume yang akan diganti.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].BlockDeviceMappings
```

3. Lepaskan volume yang akan diganti dari instans. Gunakan perintah detach-volume.

```
aws ec2 detach-volume --volume-id vol-xxxxxxxxxxxxxxxxx
```

4. Lampirkan volume pengganti ke instans. Gunakan perintah attach-volume. Untuk `--volume-id`, tentukan ID volume pengganti. Untuk`--instance-id`, tentukan ID instance tempat untuk melampirkan volume. Untuk `--device`, tentukan nama perangkat yang sama yang Anda catat sebelumnya.

```
aws ec2 attach-volume \
    --volume-id vol-01234567890abcdef \
    --instance-id i-1234567890abcdef0 \
    --device /dev/sdf
```

5. Sambungkan ke instans dan pasang volume. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

PowerShell

Untuk mengganti volume

1. Buat snapshot volume baru dari snapshot. Gunakan New-EC2Volume cmdlet dengan opsi. `-SnapshotId` Untuk `-AvailabilityZone`, tentukan Zona Ketersediaan yang sama dengan instans. Perhatikan ID volume baru dalam output.

```
New-EC2Volume `
    -VolumeType gp3 `
    -SnapshotId snap-0abcdef1234567890 `
    -AvailabilityZone us-east-1a
```

2. Dapatkan nama perangkat volume yang akan diganti. Gunakan Get-EC2Instance cmdlet. Untuk `-InstanceId`, tentukan ID instans tempat mengganti volume. Perhatikan nama perangkat dan ID volume volume yang akan diganti.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.BlockDeviceMappings | `
     Format-Table DeviceName, @{Name="VolumeId";Expression={$_.Ebs.VolumeId}}
```

3. Lepaskan volume yang akan diganti dari instans. Gunakan Dismount-EC2Volume cmdlet.

```
DismountEC2Volume -VolumeId vol-xxxxxxxxxxxxxxxxx
```

4. Lampirkan volume pengganti ke instans. Gunakan Add-EC2Volume cmdlet. Untuk `-VolumeId`, tentukan ID volume pengganti. Untuk`-InstanceId`, tentukan ID instance tempat untuk melampirkan volume. Untuk `-Device`, tentukan nama perangkat yang sama yang Anda catat sebelumnya.

```
Add-EC2Volume`
    -VolumeId vol-01234567890abcdef `
    -InstanceId i-1234567890abcdef0 `
    -Device /dev/sdf
```

5. Sambungkan ke instans dan pasang volume. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

Pemeriksaan status volume Amazon EBS

Pemeriksaan status volume memungkinkan Anda untuk lebih memahami, melacak, dan mengelola potensi ketidakkonsistenan data di volume Amazon EBS. Panduan ini dirancang untuk memberikan informasi yang Anda perlukan untuk menentukan apakah volume Amazon EBS terganggu, dan membantu Anda mengendalikan cara penanganan volume yang berpotensi tidak konsisten.

Pemeriksaan status volume adalah uji otomatis yang berjalan setiap 5 menit dan mengembalikan status lulus atau gagal. Jika semua pemeriksaan berhasil, status volume adalah `ok`. Jika pemeriksaan gagal, status volume adalah `impaired`. Jika statusnya `insufficient-data`, pemeriksaan mungkin masih berlangsung pada volume. Anda dapat melihat hasil pemeriksaan status volume untuk mengidentifikasi volume yang terganggu dan mengambil tindakan yang diperlukan.

Ketika Amazon EBS menentukan bahwa data volume berpotensi tidak konsisten, defaultnya adalah menonaktifkan volume dari inst I/O ans EC2 yang terlampir, yang membantu mencegah kerusakan data. Setelah I/O dinonaktifkan, pemeriksaan status volume berikutnya gagal, dan status volume adalah`impaired`. Selain itu, Anda akan melihat peristiwa yang memberi tahu Anda I/O bahwa dinonaktifkan, dan bahwa Anda dapat menyelesaikan status volume yang rusak dengan I/O mengaktifkan volume. EBS menunggu hingga Anda mengaktifkan I/O untuk memberi Anda kesempatan untuk memutuskan apakah akan terus membiarkan instans Anda menggunakan volume, atau menjalankan pemeriksaan konsistensi menggunakan perintah, seperti fsck (instans Linux) atau chkdsk (instans Windows), sebelum melakukannya.

> **Note**
>
> Status volume didasarkan pada pemeriksaan status volume, dan tidak mencerminkan status volume. Oleh karena itu, status volume tidak menunjukkan volume dalam `error` status (misalnya, ketika volume tidak dapat diterima I/O.) Untuk informasi tentang status volume, lihatStatus volume.

Jika konsistensi volume tertentu tidak menjadi perhatian, dan Anda lebih suka volume tersedia segera jika terganggu, Anda dapat mengganti perilaku default dengan mengonfigurasi volume untuk mengaktifkan I/O secara otomatis. Jika Anda mengaktifkan atribut volume Auto-Enable IO (`autoEnableI0`di API), pemeriksaan status volume terus berlalu. Selain itu, Anda akan melihat peristiwa yang memberi tahu Anda bahwa volume ditentukan berpotensi tidak konsisten, tetapi secara otomatis I/O diaktifkan. Ini memungkinkan Anda memeriksa konsistensi volume atau menggantinya di lain waktu.

Pemer I/O iksaan status kinerja membandingkan kinerja volume aktual dengan kinerja volume yang diharapkan. Hal ini akan memperingatkan Anda jika volume beperforma di bawah harapan. Pemeriksaan status ini hanya tersedia untuk SSD IOPS yang Tersedia (`io1` dan `io2`) dan volume SSD Tujuan Umum (`gp3`) yang terlampir pada suatu instans. Pemeriksaan status tidak valid untuk SSD Tujuan Umum (`gp2`), HDD Throughput yang Dioptimalkan (`st1`), HDD Cold (`sc1`), atau volume Magnetik (`standard`). Pemer I/O iksaan status kinerja dilakukan setiap menit sekali, dan CloudWatch mengumpulkan data ini setiap 5 menit. Diperlukan waktu hingga 5 menit sejak Anda melampirkan `io2` volume `io1` atau ke instans agar pemeriksaan status melaporkan status I/O kinerja.

> **Important**
>
> Saat menginisialisasi volume SSD IOPS yang disediakan yang dipulihkan dari snapshot, kinerja volume mungkin turun di bawah 50 persen dari level yang diharapkan, yang menyebabkan volume menampilkan status dalam pemeriksaan status `warning` Kinerja. I/O Hal ini wajar, dan Anda dapat mengabaikan status `warning` pada volume SSD IOPS yang Tersedia saat Anda menginisialisasinya. Untuk informasi selengkapnya, lihat Secara manual menginisialisasi volume setelah pembuatan.

Tabel berikut mencantumkan status untuk volume Amazon EBS.

Status volume	I/O status diaktifkan	I/O status kinerja (**io1**,**io2**, dan **gp3** volume saja)
ok	Diaktifkan (I/O Diaktifkan atau I/O Auto-Enabled)	Normal (Performa volume sesuai dengan yang diharapkan)
warning	Diaktifkan (I/O Diaktifkan atau I/O Auto-Enabled)	Terdegradasi (Performa volume di bawah ekspektasi) Sangat Menurun (Performa volume jauh di bawah harapan)
impaired	Diaktifkan (I/O Diaktifkan atau I/O Auto-Enabled) Dinonaktifkan (Volume sedang offline dan menunggu pemulihan, atau menunggu Anda untuk mengaktifkan I/O)	Terhenti (Performa volume sangat terpengaruh) Tidak Tersedia (Tidak dapat menentukan I/O kinerja karena dinon I/O aktifkan)
insufficient-data	Diaktifkan (I/O Diaktifkan atau I/O Auto-Enabled) Data Tidak Cukup	Data Tidak Cukup

Console

Untuk melihat pemeriksaan status

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

   Kolom Status Volume menampilkan status operasional setiap volume.
3. Untuk melihat detail status instans tertentu, pilih instans di kisi lalu pilih tab Pemeriksaan status.

4. Jika Anda memiliki volume dengan pemeriksaan status gagal (statusnya adalah `impaired`), lihat Bekerja dengan volume Amazon EBS yang terganggu.

Atau, Anda dapat memilih Peristiwa di navigator guna melihat semua peristiwa untuk instans dan volume Anda. Untuk informasi selengkapnya, lihat Acara volume Amazon EBS.

AWS CLI

Untuk melihat informasi status volume

Gunakan perintah describe-volume-status.

```
aws ec2 describe-volume-status --volume-ids vol-01234567890abcdef
```

Gunakan contoh berikut untuk mengidentifikasi volume yang terganggu.

```
aws ec2 describe-volume-status --filters Name=volume-status.status,Values=impaired
```

PowerShell

Untuk melihat informasi status volume

Gunakan Get-EC2VolumeStatus cmdlet.

```
Get-EC2VolumeStatus -VolumeId vol-01234567890abcdef
```

Gunakan contoh berikut untuk mengidentifikasi volume yang terganggu.

```
Get-EC2VolumeStatus -Filter @{Name="volume-status.status"; Values="impaired"}
```

Acara volume Amazon EBS

Ketika Amazon EBS menentukan bahwa data volume berpotensi tidak konsisten, ia menon I/O aktifkan volume dari instans EC2 terlampir secara default. Hal ini menyebabkan pemeriksaan status volume gagal, dan membuat peristiwa status volume yang menunjukkan penyebab kegagalan.

Untuk mengaktifkan secara otomatis I/O pada volume dengan potensi inkonsistensi data, ubah pengaturan atribut volume Auto-Enabled IO (`autoEnableIO`di API). Untuk informasi selengkapnya tentang perubahan atribut ini, lihat Bekerja dengan volume Amazon EBS yang terganggu.

Setiap peristiwa menyertakan waktu mulai yang menunjukkan waktu di mana peristiwa terjadi, dan durasi yang menunjukkan berapa I/O lama volume dinonaktifkan. Waktu akhir ditambahkan ke acara saat I/O volume diaktifkan.

Peristiwa status volume

`Awaiting Action: Enable IO`

Data volume berpotensi tidak konsisten. I/O dinonaktifkan untuk volume sampai Anda mengaktifkannya secara eksplisit. Deskripsi peristiwa berubah menjadi IO Enabled setelah Anda mengaktifkan secara eksplisit I/O.

`IO Enabled`

I/O operasi secara eksplisit diaktifkan untuk volume ini.

`IO Auto-Enabled`

I/O operasi secara otomatis diaktifkan pada volume ini setelah peristiwa terjadi. Kami menyarankan Anda memeriksa inkonsistensi data sebelum melanjutkan penggunaan data.

`Normal`

Untuk volume `io1`, `io2`, dan `gp3` saja. Performa volume sesuai dengan yang diharapkan.

`Degraded`

Untuk volume `io1`, `io2`, dan `gp3` saja. Performa volume di bawah harapan.

`Severely Degraded`

Untuk volume `io1`, `io2`, dan `gp3` saja. Performa volume jauh di bawah harapan.

`Stalled`

Untuk volume `io1`, `io2`, dan `gp3` saja. Performa volume sangat terpengaruh.

Jika Anda memiliki volume yang dinon I/O aktifkan, lihatBekerja dengan volume Amazon EBS yang terganggu. Jika Anda memiliki volume yang kiner I/O janya di bawah normal, ini mungkin merupakan kondisi sementara karena tindakan yang telah Anda lakukan (misalnya, membuat snapshot volume selama penggunaan puncak, menjalankan volume pada instance yang tidak dapat mendukung I/O bandwidth yang diperlukan, mengakses data pada volume untuk pertama kalinya, dan seterusnya).

Console

Untuk melihat peristiwa volume Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa. Semua instans dan volume yang memiliki peristiwa dicantumkan.
3. Anda dapat memfilter berdasarkan volume untuk melihat status volume saja. Anda juga dapat memfilter jenis status tertentu.
4. Pilih volume untuk menampilkan peristiwa spesifik.

AWS CLI

Untuk melihat peristiwa volume Anda

Gunakan perintah describe-volume-status.

```
aws ec2 describe-volume-status --volume-ids vol-01234567890abcdef
```

PowerShell

Untuk melihat peristiwa volume Anda

Gunakan Get-EC2VolumeStatus cmdlet.

```
Get-EC2VolumeStatus -VolumeId vol-01234567890abcdef
```

Bekerja dengan volume Amazon EBS yang terganggu

Gunakan opsi berikut jika volume terganggu karena data volume berpotensi tidak konsisten.

Opsi

- Opsi 1: Melakukan pemeriksaan konsistensi pada volume yang terlampir pada instans
- Opsi 2: Melakukan pemeriksaan konsistensi pada volume menggunakan instans lain
- Opsi 3: Hapus volume jika Anda tidak lagi membutuhkannya

Opsi 1: Melakukan pemeriksaan konsistensi pada volume yang terlampir pada instans

Opsi paling sederhana adalah mengaktifkan I/O dan kemudian melakukan pemeriksaan konsistensi data pada volume saat volume masih terpasang ke instans Amazon EC2.

Untuk melakukan pemeriksaan konsistensi pada volume yang terpasang

1. Hentikan aplikasi apa pun dari menggunakan volume.
2. Akti I/O fkan pada volume. Gunakan salah satu metode berikut.

   Console

   I/O Untuk mengaktifkan volume

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Di panel navigasi, pilih Peristiwa.
   3. Pilih volume.
   4. Pilih Tindakan, Aktifkan I/O.

   AWS CLI

   I/O Untuk mengaktifkan volume

   Gunakan perintah enable-volume-io.

   ```
   aws ec2 enable-volume-io --volume-id vol-01234567890abcdef
   ```

   PowerShell

   I/O Untuk mengaktifkan volume

   Gunakan Enable-EC2VolumeIO cmdlet.

   ```
   Enable-EC2VolumeIO -VolumeId vol-01234567890abcdef
   ```

3. Periksa data di volume.

   a. Jalan fsck kan perintah (instance Linux) atau chkdsk (instance Windows).

b. (Opsional) Tinjau semua log aplikasi atau sistem yang tersedia untuk pesan kesalahan yang relevan.

c. Jika volume telah terganggu selama lebih dari 20 menit, Anda dapat menghubungi Pusat AWS Dukungan. Pilih Pemecahan Masalah, kemudian di kotak dialog Menyelesaikan Masalah Pemeriksaan Status, pilih Hubungi Dukungan untuk mengirimkan kasus dukungan.

Opsi 2: Melakukan pemeriksaan konsistensi pada volume menggunakan instans lain

Gunakan prosedur berikut untuk memeriksa volume di luar lingkungan produksi Anda.

> **Important**
>
> Prosedur ini dapat menyebabkan hilangnya penulisan I/Os yang ditangguhkan saat volume I/O dinonaktifkan.

Untuk melakukan pemeriksaan konsistensi pada volume secara terpisah

1. Hentikan aplikasi apa pun dari menggunakan volume.
2. Lepaskan volume dari instans Untuk informasi selengkapnya, lihat Melepaskan volume Amazon EBS dari instans Amazon EC2.
3. Akti I/O fkan pada volume. Gunakan salah satu metode berikut.

   Console

   I/O Untuk mengaktifkan volume

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Di panel navigasi, pilih Peristiwa.
   3. Pilih volume yang Anda lepas di langkah sebelumnya.
   4. Pilih Tindakan, Aktifkan I/O.

   AWS CLI

   I/O Untuk mengaktifkan volume

   Gunakan perintah enable-volume-io.

```
aws ec2 enable-volume-io --volume-id vol-01234567890abcdef
```

PowerShell

I/O Untuk mengaktifkan volume

Gunakan Enable-EC2VolumeIO cmdlet.

```
Enable-EC2VolumeIO -VolumeId vol-01234567890abcdef
```

4. Lampirkan volume ke instans lainnya. Untuk informasi selengkapnya, lihat Luncurkan instance Anda dan Melampirkan volume Amazon EBS ke instans Amazon EC2.
5. Periksa data di volume.
   a. Jalan fsck kan perintah (instance Linux) atau chkdsk (instance Windows).
   b. (Opsional) Tinjau semua log aplikasi atau sistem yang tersedia untuk pesan kesalahan yang relevan.
   c. Jika volume telah terganggu selama lebih dari 20 menit, Anda dapat menghubungi Pusat AWS Dukungan. Pilih Pemecahan Masalah, lalu di kotak dialog pemecahan masalah, pilih Hubungi Dukungan untuk mengirimkan kasus dukungan.

Opsi 3: Hapus volume jika Anda tidak lagi membutuhkannya

Jika Anda ingin menghapus volume dari lingkungan Anda, cukup hapus volume. Untuk informasi tentang menghapus volume, lihat Menghapus volume Amazon EBS.

Jika Anda memiliki snapshot baru yang mencadangkan data pada volume, Anda dapat membuat volume baru dari snapshot. Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.

Auto-enable I/O untuk volume Amazon EBS yang terganggu

Ketika Amazon EBS menentukan bahwa data volume berpotensi tidak konsisten, ia menon I/O aktifkan volume dari instans EC2 terlampir secara default. Hal ini menyebabkan pemeriksaan status volume gagal, dan membuat peristiwa status volume yang menunjukkan penyebab kegagalan. Jika konsistensi volume tertentu tidak menjadi perhatian, dan Anda lebih suka volume tersedia segera jika terganggu, Anda dapat mengganti perilaku default dengan mengonfigurasi volume untuk mengaktifkan I/O secara otomatis. Jika Anda mengaktifkan atribut volume Auto-Enabled IO (`autoEnableI0`di API), I/O antara volume dan instance secara otomatis diaktifkan kembali dan

pemeriksaan status volume akan lulus. Selain itu, Anda akan melihat peristiwa yang memberi tahu Anda bahwa volume berada dalam keadaan yang berpotensi tidak konsisten, tetapi secara otomatis I/O diaktifkan. Jika peristiwa ini terjadi, Anda harus memeriksa konsistensi volume dan menggantinya jika perlu. Untuk informasi selengkapnya, lihat Acara volume Amazon EBS.

Console

Untuk melihat atribut Auto-Enabled IO dari volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume.
3. Pilih volume dan pilih tab Pemeriksaan status.

   Bid Auto-enabled I/O ang menampilkan pengaturan saat ini (Di aktifkan atau Din onaktifkan) untuk volume yang dipilih.

Untuk memodifikasi atribut Auto-Enabled IO dari volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.
3. Pilih volume dan pilih Tindakan, Kel ola diaktifkan secara otomatis I/O.
4. I/O Untuk mengaktifkan volume yang rusak secara otomatis, pilih kotak cent Auto-enable I/O ang untuk volume yang rusak. Untuk menonaktifkan fitur, kosongkan kotak centang.
5. Pilih Perbarui.

AWS CLI

Untuk melihat atribut autoEnableIO dari volume

Gunakan perintah describe-volume-attribute.

```
aws ec2 describe-volume-attribute \
    --attribute autoEnableIO \
    --volume-id vol-01234567890abcdef
```

Berikut ini adalah output contoh.

```
{
```

```
    "AutoEnableIO": {
        "Value": true
    },
    "VolumeId": "vol-01234567890abcdef"
}
```

Untuk mengubah atribut autoEnableIO dari volume

Gunakan perintah modify-volume-attribute.

```
aws ec2 modify-volume-attribute \
    --auto-enable-io \
    --volume-id vol-01234567890abcdef
```

PowerShell

Untuk melihat atribut autoEnableIO dari volume

Gunakan Get-EC2VolumeAttribute cmdlet.

```
(Get-EC2VolumeAttribute `
    -Attribute autoEnableIO `
    -VolumeId vol-01234567890abcdef).AutoEnableIO
```

Berikut ini adalah output contoh.

```
True
```

Untuk mengubah atribut autoEnableIO dari volume

Gunakan Edit-EC2VolumeAttribute cmdlet.

```
Edit-EC2VolumeAttribute `
    -AutoEnableIO $true `
    -VolumeId vol-01234567890abcdef
```

Pengujian kesalahan pada Amazon EBS

AWS Fault Injection Service (AWS FIS) adalah layanan yang dikelola sepenuhnya yang membantu Anda melakukan eksperimen injeksi kesalahan pada AWS beban kerja Anda. Dengan

tindakan EBS AWS FIS, Anda dapat menguji bagaimana aplikasi Anda merespons kesalahan penyimpanan yang dapat mengakibatkan I/O gangguan dan penurunan kinerja pada volume Anda. Lingkungan pengujian terkontrol ini memungkinkan Anda mengamati bagaimana aplikasi Anda merespons gangguan sehingga Anda dapat mengidentifikasi kelemahan dalam arsitektur Anda dan meningkatkan ketahanan keseluruhan aplikasi Anda. Dengan menggunakan I/O tindakan jeda dan tindakan injeksi latensi, Anda dapat menguji mekanisme pemantauan dan pemulihan seperti CloudWatch alarm Amazon dan alur kerja failover, serta meningkatkan ketahanan aplikasi kritis misi Anda terhadap kesalahan penyimpanan. Untuk informasi selengkap AWS FIS nya, lihat Panduan AWS Fault Injection Service Pengguna.

Eksperimen yang tersedia

Amazon EBS saat ini mendukung dua suntikan AWS FIS kesalahan:

- Jeda injeksi I/O kesalahan
- Injeksi latensi

Pertimbangan-pertimbangan

Pertimbangan berikut berlaku untuk instans Mac:

- Semua jenis volume Amazon EBS didukung. Volume root dan volume data didukung. Volume penyimpanan instans tidak didukung.
- Volume harus dilampirkan ke inst Nitro-based ans EC2.
- Volume Anda akan melanjutkan I/O kinerja aslinya setelah percobaan selesai berdasarkan durasi. Anda juga dapat menghentikan eksperimen yang sedang berjalan sebelum selesai. Atau, Anda dapat membuat kondisi berhenti untuk menghentikan percobaan jika mencapai ambang batas yang Anda tentukan dalam CloudWatch alarm.
- Anda dapat menggunakan AWS FIS dengan volume Multi-Attach yang diaktifkan. Semua instance terlampir terpengaruh. Anda tidak dapat memilih lampiran instance volume tertentu untuk eksperimen.
- FIS saat ini tidak tersedia di Zona Lokal, Pos Terdepan, atau Zona Panjang Gelombang.
- Anda dapat menguji hingga 5 volume di Zona Ketersediaan yang sama secara bersamaan saat menentukan ARN volume di konsol.
- Anda tidak dapat menggunakan AWS FIS dengan volume yang dibuat diOutpost, di AWS Wavelength Zona, atau di Zona Lokal.

Jeda injeksi I/O kesalahan

Gunakan AWS Fault Injection Service dan I/O tindakan Jeda untuk berhenti sementara I/O antara volume Amazon EBS dan instans yang dilampirkan untuk menguji bagaimana beban kerja Anda menangani I/O interupsi.

Untuk informasi selengkap AWS FIS nya, lihat Panduan AWS Fault Injection Service Pengguna .

Pertimbangan-pertimbangan

Ingatlah pertimbangan berikut untuk menjeda volume: I/O

- Jeda I/O didukung pada semua jenis Nitro-based instance.
- Untuk menguji konfigurasi batas waktu OS Anda, tetapkan durasi percobaan sama dengan atau lebih besar dari nilai yang ditentukan untuk `nvme_core.io_timeout`. Untuk informasi selengkapnya, lihat Waktu tunggu I/O operasi NVMe untuk volume Amazon EBS.
- Jika Anda mengem I/O udi ke volume yang telah di I/O jeda, hal berikut terjadi:
  - Transisi status volume ke `impaired` dalam 120 detik. Untuk informasi selengkapnya, lihat Pemeriksaan status volume Amazon EBS.
  - Met CloudWatch rik untuk `VolumeStalledIOCheck` adalah 1 jika volume I/O dijeda selama lebih dari 60 detik. Untuk mengetahui informasi selengkapnya, lihat Metrik untuk volume Amazon EBS.
  - Met CloudWatch rik untuk panjang antrian (`VolumeQueueLength`) tidak akan nol. Alarm atau pemantauan apa pun harus memantau kedalaman antrean non-nol.
  - Met CloudWatch rik untuk `VolumeReadOps` atau `VolumeWriteOps` akan`0`, yang menunjukkan bahwa volume tidak lagi diproses I/O.

Anda dapat melakukan eksperimen dasar dari konsol Amazon EC2, atau Anda dapat melakukan eksperimen yang lebih canggih menggunakan AWS FIS konsol. Untuk informasi selengkapnya tentang melakukan eksperimen lanjutan menggunakan AWS FIS konsol, lihat Tutorial untuk AWS FIS di Panduan AWS Fault Injection Service Pengguna.

Untuk melakukan percobaan dasar menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.
3. Pilih volume yang akan dijeda I/O dan pilih Tindakan , Injeksi kesalahan, Jeda volume I/O.

4. Untuk Durasi, masukkan durasi yang akan dijeda I/O antara volume dan instance. Bidang di sebelah daftar dropdown Durasi menunjukkan durasi dalam format ISO 8601.
5. Di bagian Akses layanan, pilih peran layanan IAM AWS FIS untuk diasumsikan untuk melakukan percobaan. Anda dapat menggunakan peran default, atau peran yang sudah ada yang Anda buat. Untuk informasi selengkapnya, lihat Membuat peran IAM untuk AWS FIS eksperimen.
6. Pilih Jeda volume I/O. Saat diminta, masukkan `start` di bidang konfirmasi dan pilih Mulai percobaan.
7. Pantau kemajuan dan dampak percobaan Anda. Untuk informasi selengkapnya, lihat Memantau AWS FIS di Panduan Pengguna AWS FIS .

Injeksi latensi

Gunakan tindakan Injeksi Latensi (`aws:ebs:volume-io-latency`) AWS FIS untuk mensimulasikan peningkatan I/O latensi pada volume Amazon EBS Anda untuk menguji bagaimana aplikasi Anda merespons penurunan kinerja penyimpanan. Tindakan ini memungkinkan Anda untuk menentukan nilai latensi yang akan disuntikkan serta persentase I/O yang akan terpengaruh pada volume target. Dengan AWS FIS, Anda dapat menggunakan templat percobaan latensi yang telah dikonfigurasi sebelumnya untuk memulai pengujian pola I/O latensi berbeda yang mungkin diamati selama kesalahan penyimpanan. Template ini dirancang sebagai serangkaian skenario awal yang dapat Anda gunakan untuk memperkenalkan gangguan pada aplikasi Anda guna menguji ketahanan. Mereka tidak dirancang untuk mencakup semua jenis dampak yang dapat dialami aplikasi Anda di dunia nyata. Kami menyarankan Anda untuk menyesuaikannya untuk menjalankan beberapa pengujian berbeda berdasarkan kebutuhan kinerja aplikasi Anda. Anda dapat menyesuaikan template yang tersedia atau membuat templat percobaan baru untuk menguji persyaratan khusus aplikasi Anda.

Pre-configured templat percobaan latensi

Amazon EBS menyediakan template percobaan latensi berikut melalui Konsol EBS dan pustaka AWS FIS skenario. Anda dapat langsung menggunakan templat ini pada volume target Anda untuk menjalankan eksperimen injeksi latensi.

- Latensi Berkel anjutan — Mensimulasikan latensi konstan. Eksperimen ini menggunakan satu tindakan injeksi latensi dan memiliki durasi total 15 menit. Eksperimen ini mensimulasikan latensi persisten pada 50 persen baca I/O dan 100 persen penulisan I/O: 500 ms selama 15 menit.

- Meningkatkan Latensi — Mensimulasikan peningkatan latensi secara bertahap. Eksperimen ini menggunakan lima tindakan injeksi latensi dan memiliki durasi total 15 menit. Eksperimen ini akan mensimulasikan peningkatan latensi bertahap pada 10 persen baca I/O dan 25 persen penulisan I/O: 50 ms selama 3 menit, 200 ms selama 3 menit, 700 ms selama 3 menit, 1 detik selama 3 menit, dan 15 detik selama 3 menit.
- Latensi Intermiten — Mensimulasikan lonjakan latensi intermiten yang tajam dengan periode pemulihan di antaranya. Eksperimen ini menggunakan tiga tindakan injeksi latensi dan memiliki durasi total 15 menit. Eksperimen ini akan mensimulasikan tiga lonjakan latensi pada 0,1 persen baca dan tulis I/O: lonjakan 30 detik yang berlangsung selama 1 menit, lonjakan 10 detik yang berlangsung selama 2 menit, dan lonjakan 20 detik yang berlangsung selama 2 menit. Akan ada periode pemulihan 5 menit antara setiap lonjakan latensi.
- Mengurangi Latensi - Mensimulasikan penurunan latensi secara bertahap. Eksperimen ini menggunakan lima tindakan injeksi latensi dan memiliki durasi total 15 menit. Eksperimen ini akan mensimulasikan penurunan latensi bertahap pada 10 persen baca I/O dan tulis I/O: 20 detik selama 3 menit, 5 detik selama 3 menit, 900 ms selama 3 menit, 300 ms selama 3 menit, dan 40 ms selama 3 menit.

Sesuaikan skenario yang telah dikonfigurasi sebelumnya

Anda menyesuaikan templat yang telah dikonfigurasi sebelumnya di atas atau membuat templat percobaan baru Anda sendiri menggunakan parameter yang dapat disesuaikan berikut ini.

- `readIOPercentage`— Persentase I/O operasi baca yang latensi akan disuntikkan. Ini adalah persentase dari semua I/O operasi baca pada volume yang akan dipengaruhi oleh tindakan.

  Rentang: Min 0,1%, Maks 100%
- `readIOLatencyMilliseconds`— Jumlah latensi yang disuntikkan pada I/O operasi baca. Ini adalah nilai latensi yang akan diamati pada persentase yang ditentukan dari pembac I/O aan selama percobaan.

  Rentang: Min 1 ms (io2) /10 ms (non-io2), Maks 60 detik
- `writeIOPercentage`— Persentase I/O operasi tulis yang latensi akan disuntikkan. Ini adalah persentase dari semua I/O operasi penulisan pada volume yang akan dipengaruhi oleh tindakan.

  Rentang: Min 0,1%, Maks 100%

- `writeIOLatencyMilliseconds`— Jumlah latensi yang disuntikkan pada I/O operasi tulis. Ini adalah nilai latensi yang akan diamati pada persentase penulisan yang ditentukan I/O selama percobaan.

  Rentang: Min 1ms (io2) /10ms (non-io2), Maks 60 detik
- `duration`— Durasi di mana latensi akan disuntikkan pada persentase yang I/O dipilih.

  Rentang: Min 1 detik, Maks 12 jam

Pemantauan injeksi latensi

Anda dapat memantau dampak kinerja pada volume Anda dengan cara berikut:

- Gunakan metrik latensi rata-rata CloudWatch untuk mendapatkan I/O latensi rata-rata per menit. Untuk informasi selengkapnya, lihat Memantau volume EBS Anda menggunakan CloudWatch.
- Gunakan statistik kinerja terperinci EBS yang tersedia melalui NVMe-CLI, CloudWatch agen, dan Prometheus untuk mendapatkan latensi rata-rata per detik. I/O Metrik terperinci juga menyediakan I/O histogram latensi yang dapat Anda gunakan untuk menganalisis varians latensi pada volume Anda. Untuk informasi selengkapnya, lihat statistik kinerja terperinci NVMe.
- Gunakan Pemeriksaan status volume Amazon EBS. Saat Anda menyuntikkan I/O latensi, status volume berubah ke `warning` status.

Pertimbangan-pertimbangan

Pertimbangkan hal berikut saat menggunakan injeksi latensi EBS:

- Injeksi latensi didukung pada semua jenis Nitro-based instans, kecuali: P4d, P5, P5e, Trn2u, G6, G6f, Gr6, Gr6f, M8i,,, R8i,, I8ge,, dan. M8i-flex C8i-flex R8i-flex Mac-m4pro Mac-m4
- Anda mungkin melihat varians hingga 5 persen dalam nilai latensi yang ditentukan dalam percobaan dan latensi yang dihasilkan diamati.
- Jika Anda menjalankan sejumlah kecil I/O operasi, persentase yang I/O ditentukan dalam parameter tindakan mungkin tidak sesuai dengan persentase aktual yang I/O terkena dampak tindakan.

Untuk menjalankan eksperimen injeksi latensi pada volume Amazon EBS

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

2. Pada panel navigasi, pilih Volume.
3. Pilih volume yang akan digunakan untuk menjalankan eksperimen dan pilih Tindakan, Pengujian ketahanan, Injeksi I/O latensi volume.

   Kon AWS Fault Injection Service sol terbuka.
4. Di jendela Buat eksperimen, pilih jenis eksperimen yang akan dijalankan: Intermiten, Meningkat, Berkel anjutan , atau Menur unkan.
5. Untuk pemilihan peran IAM, pilih Buat peran baru untuk membuat peran baru yang AWS FIS akan digunakan untuk melakukan eksperimen atas nama Anda. Atau, pilih Gunakan peran IAM yang ada jika sebelumnya Anda membuat peran IAM dengan izin yang diperlukan.
6. Bagian Per kiraan harga memberi Anda perkiraan biaya menjalankan eksperimen. Dengan AWS FIS, Anda dikenakan biaya per menit tindakan yang dijalankan, dari awal hingga akhir, berdasarkan jumlah akun target untuk eksperimen Anda.
7. Pilih Mulai percobaan.

Snapshot Amazon EBS

Anda dapat mencadangkan data pada volume Amazon EBS Anda dengan membuat salinan titik waktu, yang dikenal sebagai snapshot Amazon EBS. Snapshot merupakan pencadangan inkremental, yang berarti bahwa kami hanya menyimpan blok di perangkat yang telah berubah sejak snapshot terbaru Anda. Hal ini meminimalkan waktu yang diperlukan untuk membuat snapshot dan menghemat biaya penyimpanan dengan tidak menduplikasi data.

> **Important**
>
> AWS tidak secara otomatis mencadangkan data yang disimpan di volume EBS Anda. Untuk ketahanan data dan pemulihan bencana, Anda bertanggung jawab untuk membuat snapshot EBS secara teratur, atau menyiapkan pembuatan snapshot otomatis dengan menggunakan Mengotomatiskan pencadangan dengan Amazon Data Lifecycle Manager atau AWS Backup.

Snapshot disimpan di Amazon S3, di bucket S3 yang tidak dapat Anda akses secara langsung. Anda dapat membuat dan mengelola snapshot Anda menggunakan konsol Amazon EC2 atau API Amazon EC2. Anda tidak dapat mengakses snapshot menggunakan konsol Amazon S3 atau API Amazon S3.

Data snapshot direplikasi secara otomatis di semua Zona Ketersediaan di Region. Hal ini menyediakan ketersediaan serta daya tahan yang tinggi untuk data snapshot, dan memungkinkan Anda memulihkan volume di Zona Ketersediaan di Region tersebut.

Setiap snapshot berisi semua informasi yang diperlukan untuk memulihkan data Anda (dari saat ketika snapshot diambil) ke volume EBS baru. Saat Anda membuat volume EBS dari snapshot, volume baru dimulai sebagai replika persis dari volume yang digunakan untuk membuat snapshot.

Untuk informasi selengkapnya, lihat halaman produk Amazon EBS Snapshots.

Peristiwa snapshot

Anda dapat melacak status snapshot EBS Anda melalui CloudWatch Acara. Untuk informasi selengkapnya, lihat Peristiwa snapshot EBS.

Harga snapshot

Biaya untuk snapshot Anda didasarkan pada jumlah data yang disimpan. Karena snapshot bersifat inkremental, menghapus snapshot mungkin tidak mengurangi biaya penyimpanan data Anda. Data yang direferensikan secara eksklusif oleh snapshot dihapus saat snapshot dihapus, tetapi data

yang dirujuk oleh snapshot lain disimpan. Untuk informasi selengkapnya, lihat Snapshot dan volume Amazon Elastic Block Store di Panduan Pengguna AWS Billing .

Daftar Isi

- Cara kerja snapshot Amazon EBS
- Siklus hidup snapshot Amazon EBS
- Pemulihan snapshot cepat Amazon EBS
- Kunci snapshot Amazon EBS
- Blokir akses publik untuk snapshot Amazon EBS
- Snapshot lokal Amazon EBS di Outposts
- Snapshot lokal di Zona Lokal

Cara kerja snapshot Amazon EBS

Snapshot pertama yang Anda buat dari volume selalu merupakan snapshot penuh. Snapshot ini mencakup semua blok data yang ditulis ke volume pada saat membuat snapshot. Snapshot berikutnya dengan volume yang sama adalah snapshot inkremental. Snapshot ini hanya menyertakan blok data yang diubah dan baru yang ditulis ke volume sejak snapshot terakhir dibuat

Ukuran snapshot penuh ditentukan oleh ukuran data yang dicadangkan, bukan ukuran volume sumber. Demikian pula, biaya penyimpanan yang terkait dengan snapshot penuh ditentukan oleh ukuran snapshot, bukan ukuran volume sumber. Misalnya, Anda membuat snapshot pertama dari volume `200 GiB` Amazon EBS yang hanya berisi `50 GiB` data. Hal ini menghasilkan snapshot lengkap yang berukuran `50 GiB`, dan Anda ditagih untuk penyimpanan snapshot `50 GiB`.

Demikian pula, ukuran dan biaya penyimpanan snapshot tambahan ditentukan oleh ukuran data apa pun yang ditulis ke volume sejak snapshot sebelumnya dibuat. Melanjutkan contoh sebelumnya, jika Anda membuat snapshot kedua dari `200 GiB` volume yang sama setelah mengubah `20 GiB` data dan menambahkan `10 GiB` data, snapshot inkremental `30 GiB` berukuran. Anda kemudian ditagih untuk penyimpanan snapshot `30 GiB` tambahan itu.

Untuk informasi selengkapnya tentang harga snapshot, lihat Harga Amazon EBS.

> **Important**
>
> Saat Anda mengarsipkan snapshot inkremental, snapshot dikonversi ke snapshot penuh yang mencakup semua blok yang ditulis ke volume pada saat snapshot dibuat. Kemudian,

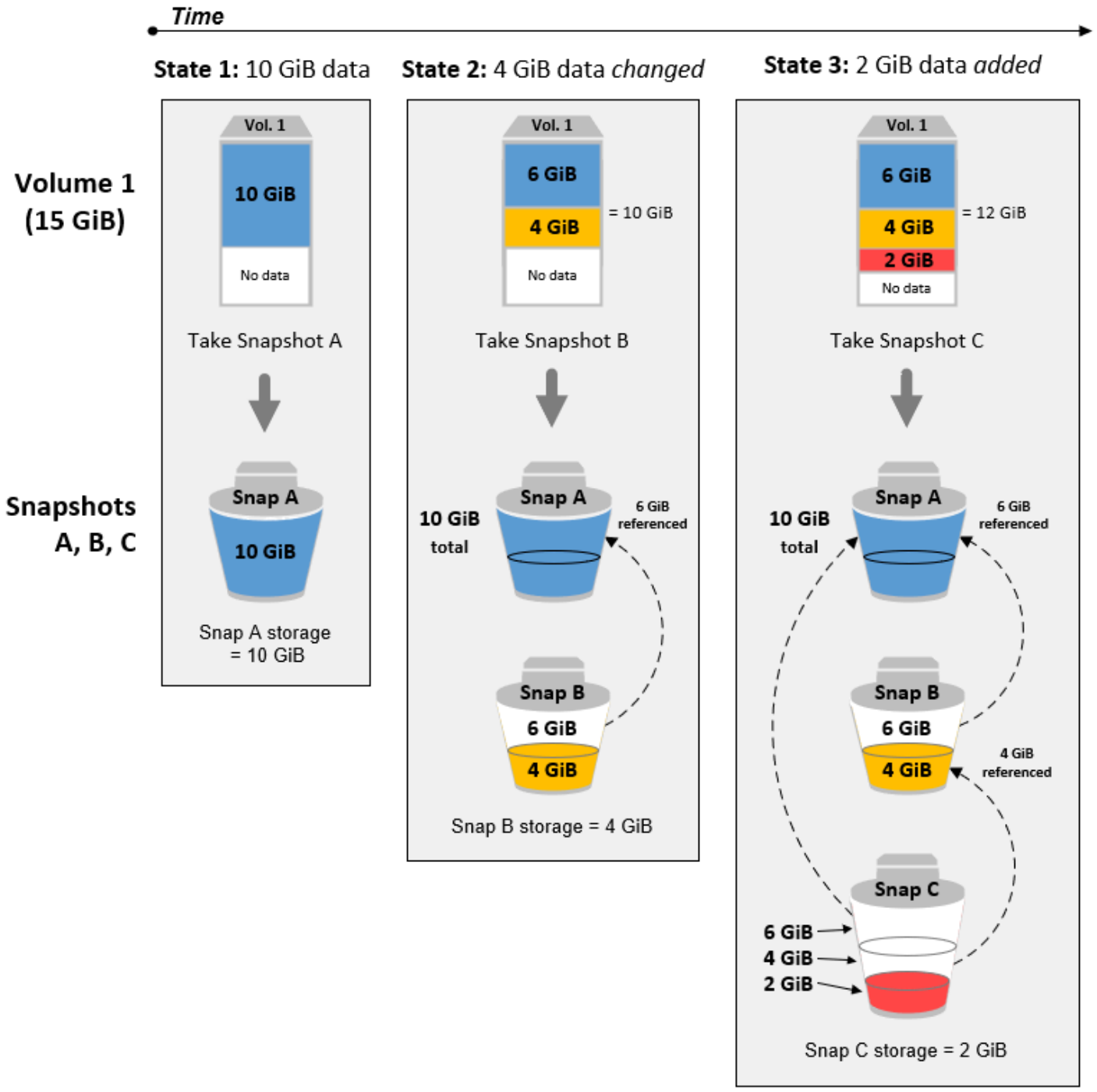
snapshot dipindahkan ke tingkat Arsip Snapshot Amazon EBS. Snapshot di tingkat arsip ditagih dengan tarif yang berbeda dari snapshot di tingkat standar. Untuk informasi selengkapnya, lihat Harga dan penagihan untuk pengarsipan snapshot Amazon EBS.

Bagian berikut menunjukkan cara snapshot EBS menangkap status volume pada satu titik waktu, dan cara snapshot berikutnya dari volume yang berubah membuat riwayat perubahan tersebut.

Banyak snapshot dengan volume yang sama

Diagram di bagian bawah ini menampilkan Volume 1, yang berukuran 15 GiB, pada tiga titik waktu. Snapshot diambil dari setiap tiga status volume ini. Diagram secara khusus menunjukkan hal berikut:

- Dalam Status 1, volume memiliki 10 GiB data. Snap A adalah snapshot pertama yang diambil dari volume. Snap A adalah snapshot lengkap dan seluruh 10 GiB data dicadangkan.
- Di Status 2, volume masih berisi 10 GiB data, tetapi hanya 4 GiB yang berubah setelah Snap A diambil. Snap B adalah snapshot inkremental. Snapshot perlu mencadangkan 4 GiB yang berubah saja. Data lain 6 GiB yang tidak berubah, yang sudah dicadangkan di Snap A, direferensikan oleh Snap B alih-alih dicadangkan lagi. Hal ini ditunjukkan dengan panah putus-putus.
- Di Status 3, 2 GiB data telah ditambahkan ke volume, untuk total 12 GiB, setelah Snap B diambil. Snap B adalah snapshot inkremental. Snapshot perlu mencadangkan hanya 2 GiB yang ditambahkan setelah Snap B diambil. Seperti yang ditunjukkan oleh panah putus-putus, Snap C juga mereferensikan 4 GiB data yang disimpan di Snap B, dan 6 GiB data yang disimpan di Snap A.
- Total penyimpanan yang diperlukan untuk tiga snapshot adalah 16 GiB total. Masing-masing menyumbang 10 GiB untuk Snap A, 4 GiB untuk Snap B, dan 2 GiB untuk Snap C.



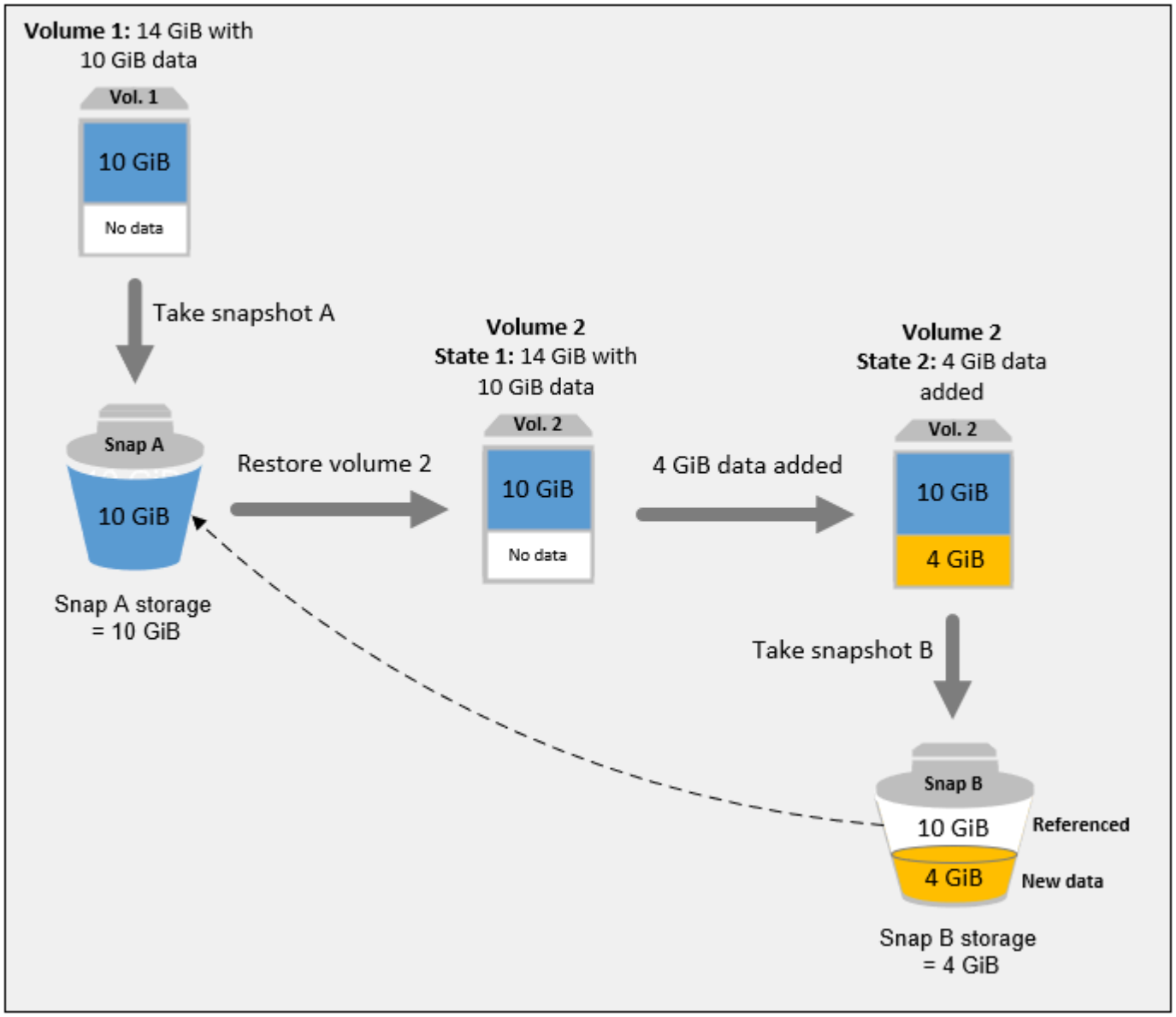
Snapshot inkremental dari volume berbeda

Diagram di bagian ini menunjukkan cara snapshot inkremental dapat diambil dari volume yang berbeda.

1. Vol 1, yang berukuran `14 GiB`, memiliki `10 GiB` data. Karena Snap A adalah snapshot pertama yang diambil dari volume, snapshot ini adalah snapshot penuh dan keseluruhan `10 GiB` data dicadangkan.
2. Vol 2 dibuat dari Snap A, sehingga ini adalah replika persis dari Vol 1 pada saat snapshot diambil.
3. Seiring waktu, `4 GiB` data ditambahkan ke Vol 2 dan ukuran total data adalah `14 GiB`.
4. Snap B diambil dari Vol 2. Untuk Snap B, hanya `4 GiB` data yang ditambahkan setelah volume dibuat dari Snap A dicadangkan. `10 GiB` data lain yang tidak berubah, yang sudah disimpan di Snap A, direferensikan oleh Snap B alih-alih dicadangkan lagi.

   Snap B adalah sebuah snapshot inkremental dari Snap A, meskipun dibuat dari volume yang berbeda.

> **Important**
>
> Diagram mengasumsikan bahwa Anda memiliki Vol 1 dan Snap A, dan bahwa Vol 2 dienkripsi dengan kunci KMS yang sama dengan Vol 1. Jika Vol 1 dimiliki oleh AWS akun lain dan akun itu mengambil Snap A dan membagikannya kepada Anda, maka Snap B akan menjadi snapshot lengkap. Atau, jika Vol 2 dienkripsi dengan kunci KMS yang berbeda dari Vol 1, Snap B akan menjadi snapshot penuh.



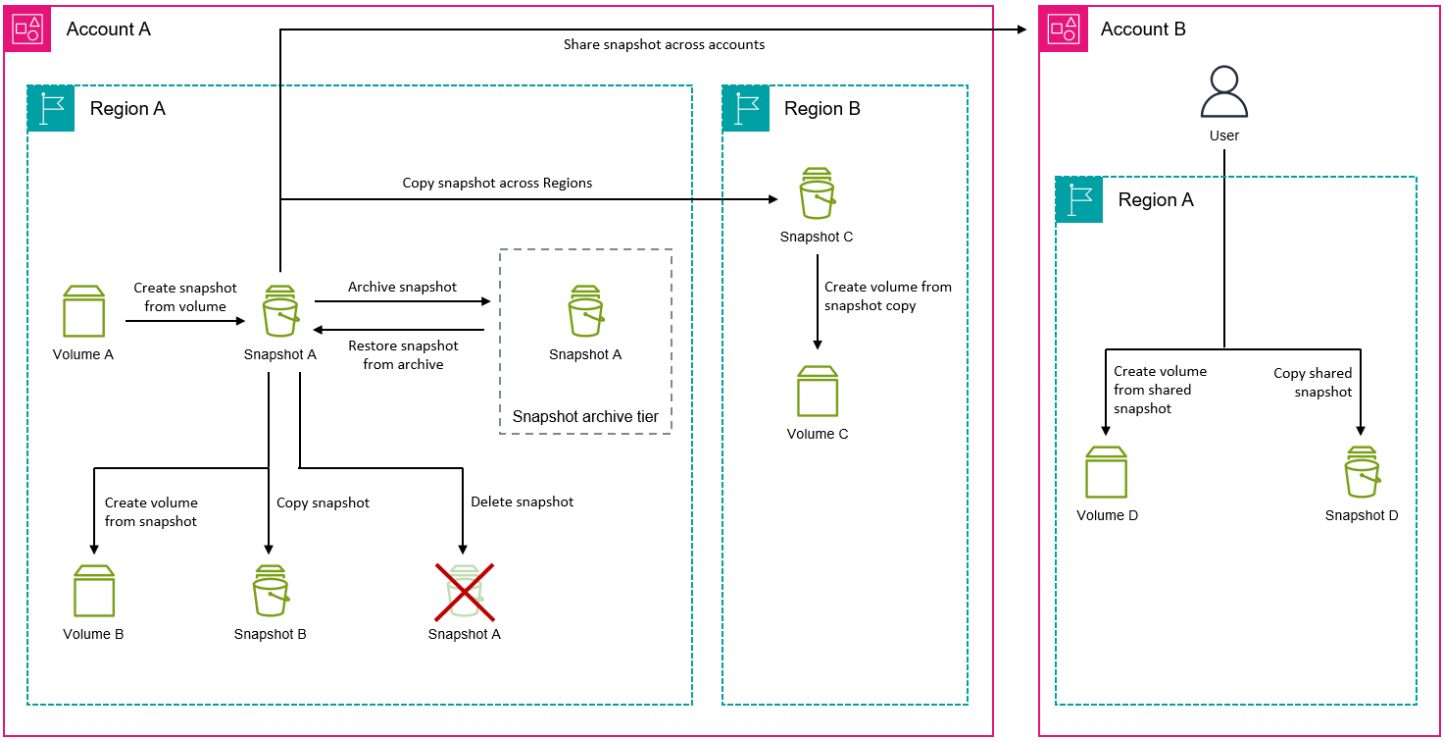
Untuk informasi selengkapnya tentang cara data dikelola saat Anda menghapus snapshot, lihat Hapus snapshot Amazon EBS.

Siklus hidup snapshot Amazon EBS

Siklus hidup snapshot Amazon EBS dimulai dengan proses pembuatan. Anda membuat snapshot dari volume Amazon EBS. Anda dapat menggunakan snapshot untuk memulihkan volume Amazon EBS baru. Anda dapat membuat salinan snapshot baik di Wilayah yang sama, atau di Wilayah yang berbeda. Anda dapat berbagi snapshot dengan orang lain Akun AWS, baik secara publik maupun pribadi. Akun tersebut dapat memulihkan volume dari snapshot bersama, atau mereka dapat

membuat salinan snapshot bersama di akun mereka sendiri. Jika Anda tidak memerlukan akses langsung ke snapshot, Anda dapat mengarsipkannya untuk menghemat biaya penyimpanan.

Gambar berikut menunjukkan tindakan yang dapat Anda lakukan pada snapshot sebagai bagian dari siklus hidup snapshot.



Tugas

- Membuat snapshot Amazon EBS
- Melihat informasi snapshot Amazon EBS
- Menyalin snapshot Amazon EBS
- Bagikan snapshot Amazon EBS dengan orang lain AWS rekening
- Mengarsipkan snapshot Amazon EBS
- Hapus snapshot Amazon EBS

Membuat snapshot Amazon EBS

Anda dapat membuat snapshot Amazon EBS dari volume Amazon EBS untuk membuat cadangan point-in-time volume tersebut. Anda dapat membuat snapshot dari masing-masing volume Amazon EBS, atau Anda dapat membuat snapshot multi-volume dari semua, atau subset, volume yang dilampirkan ke instans Amazon EC2.

Pembuatan snapshot tidak sinkron. Snapshot dibuat segera, tetapi tetap dalam `pending` keadaan sampai semua data telah ditransfer ke Amazon S3. Ini bisa memakan waktu beberapa jam untuk menyelesaikannya, tergantung pada jumlah blok yang dimodifikasi pada volume. Anda dapat terus menggunakan volume selama waktu ini tanpa memengaruhi snapshot. Snapshot hanya mencakup data yang ditulis ke volume pada saat snapshot diminta. Itu tidak termasuk data yang telah di-cache oleh aplikasi atau sistem operasi.

> **Tip**
>
> Untuk memastikan bahwa snapshot konsisten dan lengkap, sebaiknya jeda penulisan ke volume sebelum membuat snapshot. Jika Anda tidak dapat menjeda penulisan ke volume, sebaiknya lepaskan volume, dari dalam instans, sebelum membuat snapshot. Anda dapat me-mount ulang dan melanjutkan penulisan setelah snapshot memasuki `pending` status. Jika Anda membuat snapshot volume yang berfungsi sebagai perangkat root untuk instans Amazon EC2, sebaiknya hentikan instans sebelum mengambil snapshot.

Daftar Isi

- Enkripsi Snapshot
- Tujuan snapshot
- Mengotomatiskan snapshot
- Pertimbangan untuk membuat snapshot
- Membuat snapshot volume EBS
- Membuat snapshot EBS multi-volume dari instans EC2

Enkripsi Snapshot

Sebuah snapshot secara otomatis mendapatkan status enkripsi yang sama dengan volume dari mana ia dibuat. Snapshot yang dibuat dari volume yang tidak dienkripsi tidak dienkripsi. Snapshot yang dibuat dari volume terenkripsi secara otomatis dienkripsi menggunakan kunci KMS yang sama dengan volume.

> **Tip**
>
> Jika Anda perlu membuat snapshot terenkripsi dari volume yang tidak dienkripsi, pertama-tama buat snapshot volume yang tidak dienkripsi, lalu buat salinan terenkripsi dari snapshot tersebut.

Tujuan snapshot

Lokasi sumber daya sumber (volume atau instance) menentukan di mana Anda dapat membuat snapshot.

- Jika sumber daya sumber berada di Wilayah, Anda harus membuat snapshot di Wilayah yang sama dengan sumber daya sumber.
- Jika sumber daya sumber berada di Zona Lokal, Anda dapat membuat snapshot di Zona Lokal yang sama atau di Wilayah induknya. Untuk informasi selengkapnya, lihat Snapshot lokal di Zona Lokal.
- Jika sumber daya sumber ada diOutpost, Anda dapat membuat snapshot di Wilayah yang sama Outpost atau di wilayah induknya. Untuk informasi selengkapnya, lihat Snapshot lokal Amazon EBS di Outposts.

Mengotomatiskan snapshot

Anda dapat mengotomatiskan pembuatan snapshot menggunakan Amazon Data Lifecycle Manager dan AWS Backup.

Pertimbangan untuk membuat snapshot

- Sebaiknya jangan membuat snapshot volume yang dilampirkan ke instans Amazon EC2 yang hibernasi atau yang diaktifkan untuk hibernasi. Untuk informasi selengkapnya, lihat Cara kerja hibernasi instans Amazon EC2.
- Meskipun Anda dapat mengambil snapshot volume saat snapshot sebelumnya dari volume tersebut berada dalam `pending` status, memiliki beberapa snapshot dalam `pending` status untuk volume yang sama dapat mengakibatkan penurunan kinerja volume hingga snapshot selesai.
- Ada batasan jumlah snapshot yang dapat Anda miliki di `pending` negara bagian, dan jumlah snapshot bersamaan yang dapat Anda minta per jenis volume. Untuk informasi selengkapnya, lihat

Kuota untuk Amazon EBS. Jika Anda melebihi salah satu kuota ini, tunggu hingga snapshot saat ini selesai, lalu coba lagi.

Membuat snapshot volume EBS

Anda dapat membuat snapshot tunggal dari satu volume. Atau, untuk mengotomatiskan pembuatan snapshot, gunakan Amazon Data Lifecycle Manager atau AWS Backup. Untuk membuat snapshot dari semua volume untuk instans EC2, gunakan snapshot multi-volume.

Console

Untuk membuat snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot, Buat snapshot.
3. Untuk Jenis sumber daya, pilih Volume.
4. Untuk ID Snapshot, pilih snapshot yang akan digunakan untuk membuat volume. Bid ang Enkripsi menunjukkan volume dan status enkripsi snapshot yang dihasilkan. Itu tidak bisa dimodifikasi.
5. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat untuk snapshot.
6. Jika volume berada di Outpost atau di Zona Lokal, bidang tujuan Snapshot akan muncul. Lakukan salah satu tindakan berikut:
   - Jika volume berada di Zona Lokal, pilih Zona Lokal untuk membuat snapshot di Zona Lokal yang sama, atau pilih Wil AWS ayah untuk membuat snapshot di Wilayah induk Zona Lokal.
   - Jika volume ada padaOutpost, pilih AWS Outpost, untuk membuat snapshot yang samaOutpost, atau pilih Wil AWS ayah untuk membuat snapshot di Wilayah indukOutpost.

   > **Note**
   >
   > Jika volume berada di Wilayah, tujuan Snapshot tidak muncul. Snapshot secara otomatis dibuat di Wilayah yang sama dengan volume.

7. (Opsional) Untuk menetapkan tag khusus ke snapshot, di bagian Tag, pilih Tam bah tag, lalu masukkan pasangan kunci-nilai. Anda dapat menambahkan hingga 50 tanda.

8. Pilih Buat snapshot.

AWS CLI

Untuk membuat snapshot

Menggunakan perintah create-snapshot.

```
aws ec2 create-snapshot  \
    --volume-id vol-01234567890abcdef \
    --description "Snapshot of the root volume for i-1234567890abcdef0"
```

PowerShell

Untuk membuat snapshot

Gunakan New-EC2Snapshot cmdlet.

```
New-EC2Snapshot `
    -VolumeId vol-01234567890abcdef `
    -Description "Snapshot of the root volume for i-1234567890abcdef0"
```

Membuat snapshot EBS multi-volume dari instans EC2

Secara default, saat Anda membuat snapshot multi-volume dari instans Amazon EC2, Amazon EBS membuat snapshot dari semua volume Amazon EBS yang dilampirkan ke instans. Namun, Anda dapat memilih untuk mengecualikan volume root, atau volume data tertentu jika diperlukan.

> **Tip**
>
> Sebaiknya beri tag snapshot multi-volume agar mudah untuk mengidentifikasi dan mengelolanya secara kolektif. Anda juga dapat menyalin tag dari volume sumber ke snapshot yang sesuai untuk mengatur metadata snapshot, seperti kebijakan akses, informasi lampiran, dan alokasi biaya, agar sesuai dengan volume sumber.

Pertimbangan untuk snapshot multi-volume

- Jika semua snapshot berhasil diselesaikan, `createSnapshots` CloudWatch acara dengan hasil `succeeded` dikirim ke AWS akun Anda. Jika salah satu snapshot dalam kumpulan snapshot

multi-volume gagal, semua snapshot lainnya masuk ke `error` status dan `createSnapshots` CloudWatch peristiwa dengan hasil `failed` dikirim ke akun Anda. Untuk informasi selengkapnya, lihat Membuat snapshot (membuatSnapshots).

- Multi-volume snapshot mendukung hingga 128 volume Amazon EBS yang dilampirkan ke instans, termasuk volume root dan hingga 127 volume data.
- Setiap snapshot dalam set snapshot multi-volume adalah snapshot individual yang dapat digunakan dengan cara yang sama, dan yang mendukung fitur yang sama, sebagai snapshot individual.
- Anda dapat mengambil snapshot yang konsisten dengan aplikasi dari semua volume Amazon EBS yang dilampirkan ke instans Amazon EC2 Windows menggunakan dokumen perintah. AWS Systems Manager

Console

Untuk membuat snapshot multi-volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot, Buat snapshot.
3. Untuk Tipe sumber daya, pilih Instans.
4. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat untuk snapshot tersebut. Deskripsi ini diterapkan ke semua snapshot.
5. Jika instance berada di Outpost atau di Zona Lokal, bidang tujuan Snapshot akan muncul. Lakukan salah satu tindakan berikut:
   - Jika instance berada di Zona Lokal, pilih Zona Lokal untuk membuat snapshot di Zona Lokal yang sama, atau pilih Wil AWS ayah untuk membuat snapshot di Wilayah induk Zona Lokal.
   - Jika instance berada diOutpost, pilih AWS Outpost, untuk membuat snapshot yang samaOutpost, atau pilih Wil AWS ayah untuk membuat snapshot di Wilayah induk. Outpost

> **Note**
>
> Jika instance berada di Wilayah, tujuan Snapshot tidak muncul. Snapshot secara otomatis dibuat di Wilayah yang sama dengan instance.

6. (Opsional) Untuk mengecualikan volume root instans, pilih K ecualikan volume root.
7. (Opsional) Untuk mengecualikan volume data, pilih K ecualikan volume data tertentu. Bagian Volume data terlampir mencantumkan semua volume data yang saat ini dilampirkan ke instans yang dipilih.

   Pilih volume data yang akan dikecualikan. Hanya volume yang tetap tidak dipilih yang akan disertakan dalam kumpulan snapshot multivolume.
8. (Opsional) Untuk menyalin tag secara otomatis dari volume sumber ke snapshot yang sesuai, untuk Salin tag dari volume sumber, pilih Salin tag.
9. (Opsional) Untuk menetapkan tag khusus tambahan ke snapshot, di bagian Tag, pilih Tam bah tag, lalu masukkan pasangan kunci-nilai. Anda dapat menambahkan hingga 50 tanda.
10. Pilih Buat snapshot.

AWS CLI

Untuk membuat snapshot multi-volume

Gunakan perintah create-snapshots.

```
aws ec2 create-snapshots \
    --instance-specification InstanceId=i-1234567890abcdef0 \
    --description "from a multi-volume snapshot of i-1234567890abcdef0"
```

Untuk mengecualikan volume root, tambahkan yang berikut ini ke `--instance-specification` opsi.

```
ExcludeBootVolume=true
```

Untuk mengecualikan volume data, tambahkan yang berikut ke `--instance-specification` opsi.

```
ExcludeDataVolumeIds=vol-01234567890abcdef
```

PowerShell

Untuk membuat snapshot multi-volume

Gunakan New-EC2SnapshotBatch cmdlet.

```
New-EC2SnapshotBatch `
    -InstanceSpecification_InstanceId i-1234567890abcdef0 `
    -Description "from a multi-volume snapshot of i-1234567890abcdef0"
```

Untuk mengecualikan volume root, tambahkan parameter berikut.

```
-InstanceSpecification_ExcludeBootVolume $true
```

Untuk mengecualikan volume data, tambahkan parameter berikut.

```
-InstanceSpecification_ExcludeDataVolumes "vol-01234567890abcdef"
```

Melihat informasi snapshot Amazon EBS

Anda dapat melihat informasi terperinci tentang snapshot Anda.

Console

Untuk melihat informasi snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Untuk melihat snapshot yang Anda miliki, di sudut kiri atas layar, pilih Dimiliki oleh saya. Anda juga dapat memfilter daftar snapshot menggunakan tanda dan atribut snapshot lainnya. Di bidang Filter, pilih bidang atribut, lalu pilih atau masukkan nilai atribut. Misalnya, untuk hanya melihat snapshot terenkripsi, pilih Enkripsi, lalu masukkan `true`.
4. Untuk melihat informasi selengkapnya tentang snapshot tertentu, pilih ID di daftar.

> **Note**
>
> Bidang Ukuran snapshot penuh menunjukkan ukuran penuh snapshot, dalam byte. Ini bukan ukuran inkremental dari snapshot. Sebaliknya, ini mewakili ukuran semua blok yang ditulis ke volume sumber pada saat snapshot dibuat.
> Bid ang ukuran volume menunjukkan ukuran volume EBS yang akan dibuat dari snapshot jika tidak ada ukuran lain yang ditentukan.

AWS CLI

Untuk melihat informasi snapshot

Gunakan perintah describe-snapshots.

Example Contoh 1: Filter berdasarkan tanda

Contoh berikut menjelaskan snapshot dengan tag stack=production.

```
aws ec2 describe-snapshots --filters Name=tag:Stack,Values=production
```

Example Contoh 2: Filter berdasarkan volume

Contoh berikut menjelaskan snapshot yang dibuat dari volume yang ditentukan.

```
aws ec2 describe-snapshots --filters Name=volume-id,Values=vol-049df61146c4d7901
```

Example Contoh 3: Memfilter berdasarkan usia snapshot

Anda dapat menggunakan JMesPath untuk memfilter hasil menggunakan ekspresi. Misalnya, perintah berikut menampilkan ID semua snapshot yang dibuat oleh akun Anda sebelum tanggal yang ditentukan. Jika Anda tidak menentukan pemiliknya, hasilnya akan menyertakan semua snapshot publik.

```
aws ec2 describe-snapshots \
    --filters Name=owner-id,Values=123456789012 \
    --query "Snapshots[?(StartTime<='2024-03-31')].[SnapshotId]" \
    --output text
```

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>='2024-01-01') && (StartTime<='2024-12-31')].
[SnapshotId]" \
    --output text
```

PowerShell

Untuk melihat informasi snapshot

Gunakan Get-EC2Snapshot cmdlet.

Contoh 1: Jelaskan snapshot

Contoh berikut menjelaskan snapshot tertentu.

```
Get-EC2Snapshot -SnapshotId snap-0abcdef1234567890
```

Contoh 2: Filter berdasarkan volume

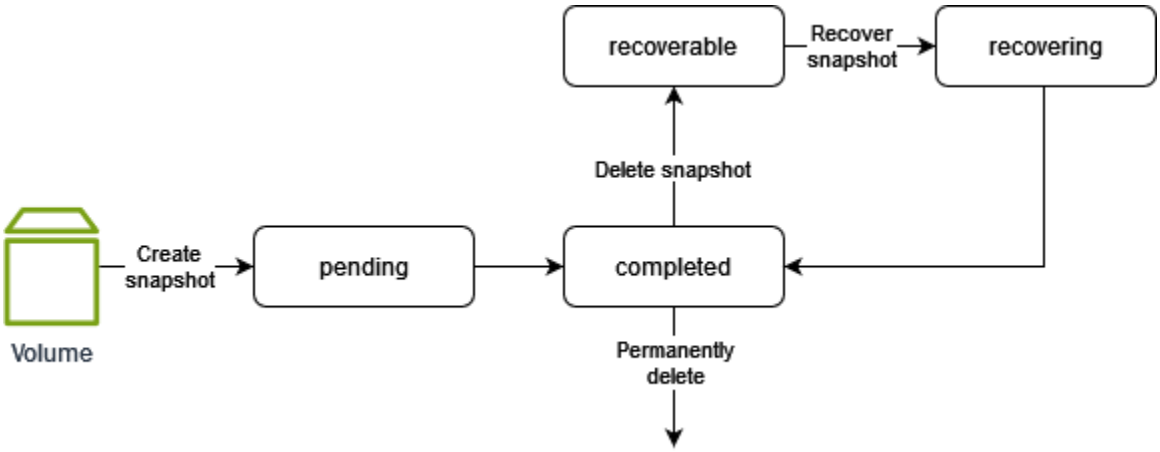
Contoh berikut menjelaskan snapshot yang dibuat dari volume yang ditentukan.

```
Get-EC2Snapshot`
    -Filter @{Name="volume-id"; Values="vol-01234567890abcdef"}
```

Status snapshot

Transisi snapshot Amazon EBS melalui status yang berbeda dari saat dibuat hingga dihapus secara permanen.

Ilustrasi berikut menunjukkan transisi antara status snapshot. Saat Anda membuat snapshot, ia memasuki `pending` status. Setelah snapshot siap digunakan, ia memasuki `completed` negara bagian. Ketika Anda telah memutuskan bahwa Anda tidak lagi memerlukan snapshot, Anda dapat menghapusnya. Jika Anda menghapus snapshot yang cocok dengan aturan retensi Recycle Bin, snapshot tersebut akan disimpan di Recycle Bin dan masuk ke `recoverable` status. Jika Anda memulihkan snapshot dari Recycle Bin, ia memasuki `recovering` status dan kemudian `completed` status. Jika tidak, itu dihapus secara permanen.



Tabel berikut merangkum status snapshot.

Status	Deskripsi
`pending`	Proses pembuatan snapshot masih berlangsung. Snapshot tidak dapat digunakan saat berada di `pending` negara bagian.
`completed`	Proses pembuatan snapshot telah selesai dan snapshot siap digunakan.
`recoverable`	Snapshot saat ini ada di Recycle Bin. Untuk menggunakan snapshot, Anda harus memulihkannya terlebih dahulu dari Recycle Bin.
`recovering`	Snapshot sedang dipulihkan dari Recycle Bin. Setelah snapshot dipulihkan, ia beralih ke `completed` status dan siap digunakan.
`error`	Proses pembuatan snapshot gagal. Snapshot tidak dapat digunakan jika berada di `error` negara bagian.

Menyalin snapshot Amazon EBS

Setelah Anda membuat snapshot, dan telah mencapai `completed` status, Anda dapat membuat salinannya. Salinan snapshot adalah salinan persis dari aslinya, tetapi memiliki ID sumber daya yang unik. Anda dapat menyalin snapshot yang Anda miliki dan snapshot yang dibagikan dengan Anda, secara pribadi atau publik. Anda mungkin perlu menyalin snapshot untuk kasus penggunaan berikut:

- Ekspansi geografis — Anda perlu meluncurkan aplikasi Anda di Wilayah baru.
- Migrasi — Anda perlu memindahkan aplikasi ke tujuan baru, untuk mengaktifkan ketersediaan yang lebih baik atau untuk meminimalkan biaya.
- Pemulihan bencana — Anda perlu mencadangkan data dan log ke Wilayah sekunder untuk tujuan redundansi data.
- Enkripsi — Anda perlu mengenkripsi snapshot yang sebelumnya tidak dienkripsi atau mengenkripsi ulang snapshot terenkripsi menggunakan kunci KMS yang berbeda.

- Salin snapshot bersama — Anda perlu menyalin snapshot yang dibagikan dengan Anda.
- Persyaratan penyimpanan dan audit data — Anda perlu menyalin snapshot terenkripsi dari satu AWS akun ke akun lain untuk menyimpan data untuk audit atau penyimpanan data. Menggunakan akun yang berbeda melindungi Anda jika AWS akun utama Anda dikompromikan.

Untuk menyalin snapshot multi-volume, identifikasi semua snapshot yang merupakan bagian dari kumpulan itu menggunakan tag yang Anda tetapkan selama pembuatan, lalu salin satu per satu.

Untuk informasi tentang menyalin snapshot Amazon RDS, lihat Menyalin Snapshot DB dalam Panduan Pengguna Amazon RDS.

Harga

Untuk informasi harga tentang menyalin snapshot, lihat Harga Amazon EBS.

Daftar Isi

- Pertimbangan untuk menyalin snapshot
- Tujuan untuk salinan snapshot
- Penyalinan snapshot inkremental
- Time-based salinan untuk snapshot dan AMI Amazon EBS EBS-backed
- Enkripsi dan penyalinan snapshot
- Menyalin snapshot

Pertimbangan untuk menyalin snapshot

- Anda dapat menyalin AWS Marketplace, snapshot VM Import/Export, dan Storage Gateway, tetapi Anda harus memverifikasi bahwa snapshot didukung di Wilayah tujuan.
- Ada batas permintaan salinan snapshot `20` bersamaan per tujuan. Jika melebihi kuota ini, Anda menerima `ResourceLimitExceeded` kesalahan. Jika Anda menerima kesalahan ini, tunggu satu atau beberapa permintaan salinan selesai sebelum membuat permintaan salinan snapshot baru.
- User-defined tag tidak disalin dari snapshot sumber ke salinan snapshot. Anda dapat menambahkan tanda yang ditentukan pengguna selama atau setelah operasi penyalinan.
- Snapshot yang dibuat oleh operasi penyalinan snapshot memiliki ID volume arbitrer, seperti `vol-ffff` atau `vol-ffffffff`. ID volume yang tidak boleh digunakan untuk tujuan apa pun.

- Resource-level izin yang ditentukan untuk operasi penyalinan snapshot dapat diterapkan ke salinan snapshot dan snapshot sumber. Sebagai contoh, lihat Contoh: Menyalin snapshot.
- Jika Anda menyalin snapshot yang diaktifkan untuk pemulihan snapshot cepat, salinan snapshot tidak secara otomatis diaktifkan untuk pemulihan snapshot cepat. Anda harus secara eksplisit mengaktifkan pemulihan snapshot cepat untuk salinan snapshot.
- Jika Anda menyalin snapshot dan mengenkripsinya ke kunci KMS baru, salinan lengkap (tidak inkremental) dibuat. Hal ini menyebabkan biaya penyimpanan tambahan.
- Jika Anda menyalin snapshot ke Wilayah baru, salinan lengkap (non-inkremental) dibuat. Hal ini menyebabkan biaya penyimpanan tambahan.
- Jika Anda menggunakan transfer data eksternal atau lintas wilayah, biaya transfer data EC2 tambahan akan berlaku. Jika Anda menghapus snapshot apa pun setelah inisiasi, Anda masih dikenakan biaya untuk data yang telah ditransfer.

Tujuan untuk salinan snapshot

Lokasi snapshot sumber menentukan apakah Anda dapat menyalinnya atau tidak.

- Jika snapshot sumber berada di Wilayah, Anda dapat menyalinnya di dalam Wilayah itu, ke Wilayah lain, ke yang Outpost terkait dengan Wilayah itu, atau ke Zona Lokal di Wilayah tersebut.
- Jika snapshot sumber berada di Zona Lokal, Anda dapat menyalinnya dalam Zona Lokal yang sama, ke Zona Lokal lain di grup zona yang sama, atau ke Wilayah induk Zona Lokal tersebut.
- Jika snapshot sumber ada diOutpost, Anda tidak dapat menyalinnya.

Penyalinan snapshot inkremental

Operasi penyalinan snapshot dalam akun dan Wilayah yang sama menggunakan kunci KMS yang sama selalu merupakan salinan tambahan. Namun, jika Anda mengenkripsi salinan snapshot menggunakan kunci KMS yang berbeda, salinannya adalah salinan lengkap.

Saat Anda menyalin snapshot di seluruh Wilayah atau akun, salinan tersebut adalah salinan inkremental jika syarat berikut terpenuhi:

- Snapshot disalin ke Wilayah atau akun tujuan sebelumnya.
- Salinan snapshot terbaru masih ada di Wilayah atau akun tujuan.
- Salinan snapshot terbaru belum diarsipkan.

- Semua salinan snapshot di Wilayah atau akun tujuan tidak dienkripsi atau dienkripsi menggunakan kunci KMS yang sama.

> **Tip**
>
> Sebaiknya beri tag salinan snapshot dengan ID volume dan waktu pembuatan sehingga Anda dapat melacak salinan snapshot terbaru dari volume di Wilayah atau akun tujuan.

Untuk melihat apakah salinan snapshot Anda bersifat inkremental, periksa peristiwa Copy CloudWatch Snapshot.

Time-based salinan untuk snapshot dan AMI Amazon EBS EBS-backed

Time-based salinan dapat membantu Anda memenuhi kepatuhan atau persyaratan bisnis untuk replikasi data dengan memastikan bahwa snapshot EBS dan EBS-backed AMI Anda disalin, di dalam dan di seluruh Wil AWS ayah, dalam jangka waktu tertentu. Time-based salinan juga dapat membantu administrator cadangan memenuhi persyaratan pemulihan bencana yang ketat (Tujuan Titik Pemulihan dan Tujuan Waktu Pemulihan), dan meningkatkan kelincahan pengembangan dengan memastikan waktu penyalinan yang dapat diprediksi untuk snapshot dan EBS-backed AMI.

Dengan snapshot berbasis waktu dan operasi penyalinan EBS-backed AMI, Anda menentukan durasi penyelesaian, antara 15 menit dan 48 jam, di mana salinan harus diselesaikan. Durasi penyelesaian harus ditentukan dalam kenaikan 15 menit.

Topik

- Kuota
- Tentukan durasi penyelesaian Anda
- Pertimbangan-pertimbangan
- Memantau
- Harga dan penagihan

Kuota

Kuota berikut berlaku untuk snapshot berbasis waktu dan operasi pen EBS-backed yalinan AMI:

Kuota	Deskripsi	Nilai kuota	Dapat Disesuaikan
Kuota throughput operasi salinan snapshot	Throughput maksimum yang dapat dicapai dengan operasi penyalinan snapshot berbasis waktu tunggal. **Note** Untuk operasi penyalinan AMI, kuota berlaku untuk setiap snapshot individu yang terkait dengan AMI.	500 MiB/s	Tidak
Kuota throughput salinan snapshot kumulatif	Throughput kumulatif maksimum yang dapat dicapai dengan operasi penyalinan snapshot berbasis waktu bersamaan antara Wilayah sumber dan tujuan. **Note** Untuk operasi penyalinan AMI, setiap snapshot individu yang	2.000 MiB/s	Ya

Kuota	Deskripsi	Nilai kuota	Dapat Disesuaikan
	terkait dengan AMI diperhitungkan dalam kuota.		

Saat memulai operasi penyal inan snapshot berbasis waktu, Anda menentukan durasi penyelesaian. Throughput yang digunakan oleh permintaan ditentukan oleh ukuran data snapshot dan durasi penyelesaian yang diminta. Misalnya, jika Anda menyalin snapshot yang memiliki 225.000 MiB (0,214 TiB) data, dan Anda meminta durasi penyelesaian 15 menit, throughput adalah 250 MiB/s (225.000 MiB ÷ 15 menit = 250). MiB/s

Saat Anda memulai operasi pen yalinan AMI berbasis waktu, durasi penyelesaian yang Anda tentukan berlaku untuk setiap snapshot yang terkait dengan AMI. Karena setiap snapshot dapat memiliki ukuran yang berbeda, setiap snapshot disalin pada throughput yang berbeda untuk memastikan bahwa semua snapshot disalin dalam durasi penyelesaian. Misalnya, Anda memiliki AMI dengan snapshot terkait berikut:

- Cuplikan 1:200.000 MiB
- Snapshot 2:500.000 MiB
- Cuplikan 3:450.000 MiB

Jika Anda memulai salinan berbasis waktu untuk AMI ini dan menentukan durasi penyelesaian 60 menit, permintaan akan menggunakan throughput berikut:

- Cuplikan 1:55,56 MiB/s (200.000 MiB ÷ 60 menit = 55,56) MiB/s
- Snapshot 2:138,89 MiB/s (500.000 MiB ÷ 60 menit = 138,89) MiB/s
- Cuplikan 3:125 MiB/s (450.000 MiB ÷ 60 menit = 125) MiB/s

Ini berarti permintaan menggunakan 319,45 MiB/s kuota throughput salinan snapshot kumulatif Anda untuk memastikan bahwa salinan selesai dalam 60 menit.

Jika Anda memulai permintaan salinan snapshot atau EBS-backed AMI berbasis waktu dan kuota throughput salinan snapshot kumulatif yang tersedia adalah:

- lebih besar dari atau sama dengan tingkat throughput yang diperlukan, salinan selesai dalam durasi penyelesaian yang diminta.
- kurang dari tingkat throughput yang diperlukan tetapi lebih besar dari nol, permintaan berhasil tetapi akan memakan waktu lebih lama dari yang Anda minta. Salinan selesai menggunakan kuota throughput yang tersedia.
- nol (kuota tercapai), permintaan gagal.

Tentukan durasi penyelesaian Anda

Durasi penyelesaian minimum yang dapat Anda minta untuk snapshot berbasis waktu atau operasi EBS-backed penyalinan AMI adalah 15 menit, dan durasi penyelesaian maksimum yang dapat Anda minta adalah 48 jam. Durasi penyelesaian harus ditentukan dalam kenaikan 15 menit.

Operasi penyalinan snapshot berbasis waktu bersamaan

Anda dapat melakukan operasi penyalinan snapshot berbasis waktu bersamaan antara Wilayah sumber dan tujuan yang sama, selama throughput gabungan dari semua operasi bersamaan berada dalam kuota throughput salinan snapshot kumulatif Anda (2.000 secara MiB/s default).

Untuk menentukan apakah Anda dapat mencapai durasi penyelesaian yang diperlukan untuk snapshot yang ada, bagilah ukuran gabungan semua snapshot Anda dengan durasi penyelesaian yang diperlukan untuk menentukan tingkat throughput yang diperlukan.

> **Tip**
>
> Jika Anda tidak mengetahui ukuran pasti data dalam snapshot Anda, Anda dapat menggunakan ukuran snapshot penuh sebagai proxy sebagai gantinya. Untuk mendapatkan ukuran snapshot penuh, gunakan perintah `describe-snapshots` AWS CLI .

```
required throughput rate = combined snapshot size ÷ required completion duration
```

Jika tingkat throughput yang diperlukan kurang dari kuota throughput salinan snapshot kumulatif Anda, Anda dapat mencapai durasi penyelesaian yang diperlukan. Jika tingkat throughput yang diperlukan lebih besar dari kuota throughput salinan snapshot kumulatif Anda, sebaiknya Anda meminta peningkatan kuota yang setidaknya 10% lebih tinggi dari tingkat throughput yang diperlukan.

> **Tip**
>
> Konsol Amazon EC2 menyediakan kalkulator yang dapat Anda gunakan untuk memeriksa berapa banyak data snapshot yang Anda salin antara dua Wilayah selama periode tertentu, dan durasi penyelesaian minimum yang dapat dicapai yang dapat Anda capai untuk jumlah data tersebut, berdasarkan kuota throughput salinan snapshot kumulatif tertentu. Kalkulator menggunakan `SnapshotCopyBytesTransferred` CloudWatch metrik untuk menghitung data yang disalin antara dua Wilayah selama suatu periode. Untuk membuka kalkulator, di panel navigasi konsol Amazon EC2, pilih Snapshots, lalu pilih Tind akan, Luncurkan kalkul ator durasi salinan.
> Kalkulator durasi salinan snapshot tidak didukung dengan AWS Outposts, Zona Lokal, dan Zona Panjang Gelombang.

Operasi penyalinan snapshot berbasis waktu individual

Anda dapat menghitung durasi penyelesaian minimum untuk operasi penyalinan snapshot berbasis waktu individual dengan membagi ukuran data snapshot dengan kuota throughput operasi salinan snapshot (500 MiB/s).

> **Tip**
>
> Jika Anda tidak mengetahui ukuran pasti data dalam snapshot Anda, Anda dapat menggunakan ukuran snapshot penuh sebagai proxy sebagai gantinya. Untuk mendapatkan ukuran snapshot penuh, gunakan perintah describe-snapshots AWS CLI .

```
minimum completion duration = Max(15 minutes, (snapshot data size ÷ 500 MiB/s)
```

Misalnya, durasi penyelesaian minimum untuk snapshot dengan 900.000 MiB data adalah 30 menit.

```
minimum completion duration = Max(15 minutes, (900,000 MiB ÷ 500 MiB/s)
= Max(15 minutes, 30 minutes)
= 30 minutes
```

Time-based Operasi penyalinan AMI

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

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

Pertimbangan-pertimbangan

- Anda dapat memulai operasi snapshot berbasis waktu dan penyal EBS-backed inan AMI saat menyalin snapshot dalam Wilayah yang sama atau saat menyalin snapshot di seluruh Wilayah.
- Jika Anda memulai dua operasi penyalinan berbasis waktu untuk snapshot atau AMI yang sama, durasi penyelesaian operasi penyalinan kedua dimulai hanya setelah operasi penyalinan pertama selesai.
- Time-based operasi penyalinan dan kalkulator durasi salinan snapshot tidak didukung dengan AWS Outposts, Zona Lokal, dan Zona Panjang Gelombang.

Memantau

Anda dapat memantau kemajuan snapshot berbasis waktu dan operasi pen EBS-backed yalinan AMI menggunakan konsol Amazon EC2 dan. AWS CLI Di konsol, pilih snapshot dan kemudian, di tab Detail, periksa bidang Progress. Dengan AWS CLI, periksa elemen `Progress` output dalam respons perintah describe-snapshots.

Anda dapat memeriksa apakah snapshot berbasis waktu atau operasi pen EBS-backed yalinan AMI selesai dalam durasi penyelesaian yang diminta dengan memeriksa perbedaan antara waktu Dim ulai dan Selesai di konsol, atau `StartTime` dan `CompletionTime` dalam `describe-snapshots` respons.

Anda juga dapat menggunakan EventBridge acara `copySnapshot` Amazon untuk memantau hasil operasi penyalinan berbasis waktu. Peristiwa menunjukkan apakah operasi selesai dan apakah durasi penyelesaian yang diminta terpenuhi. Jika durasi penyelesaian tidak terpenuhi, acara tersebut mencakup informasi lebih lanjut tentang penyebabnya. Untuk informasi selengkapnya, lihat Peristiwa snapshot EBS.

Harga dan penagihan

> **Note**
>
> Mirip dengan operasi penyalinan snapshot standar, jika Anda menyalin snapshot ke Wilayah baru, salinan lengkap (non-inkremental) dibuat, yang menghasilkan biaya penyimpanan tambahan. Salinan berikutnya dari snapshot yang sama bersifat inkremental. Selain itu, jika Anda menggunakan transfer data eksternal atau lintas wilayah, biaya transfer data Amazon EC2 tambahan akan berlaku.

Biaya tambahan berlaku untuk operasi snapshot berbasis waktu dan pen EBS-backed yalinan AMI. Time-based operasi penyalinan dibebankan pada tarif yang didasarkan pada durasi penyelesaian yang diminta, per GiB data snapshot yang disalin. Tarif tetap adalah sebagai berikut:

> **Note**
>
> Durasi penyelesaian harus ditentukan dalam kenaikan 15 menit. Durasi penyelesaian minimum adalah 15 menit, dan maksimum adalah 48 jam.

- 15 menit - $0.020 per GiB data
- 30 menit dan 45 menit - $0.018 per GiB data
- 1 jam hingga 1 jam 45 menit - $0,016 per GiB data
- 2 jam hingga 3 jam 45 menit - $0.014 per GiB data
- 4 jam hingga 7 jam 45 menit - $0.012 per GiB data
- 8 jam hingga 15 jam 45 menit - $0.010 per GiB data
- 16 jam atau lebih - $0.005 per GiB data

Misalnya, jika Anda menyalin snapshot dengan 3.000 GiB data dengan durasi penyelesaian 8 jam, Anda akan ditagih $30 ($0,010 x 3.000 GiB).

Jika Anda memulai operasi penyalinan berdasarkan waktu, tetapi durasi penyelesaian yang diminta tidak terpenuhi karena Anda melebihi kuota, Anda akan ditagih berdasarkan durasi penyelesaian aktual, bukan durasi penyelesaian yang diminta. Misalnya, jika Anda meminta durasi penyelesaian

1 jam, tetapi operasi selesai dalam 2 jam, Anda akan ditagih berdasarkan tarif untuk durasi penyelesaian 2 jam.

Jika Amazon EBS tidak dapat mencapai durasi penyelesaian yang diminta atau jika permintaan dibatalkan karena masalah di sisi layanan, Anda tidak ditagih biaya tambahan untuk operasi penyalinan snapshot berbasis waktu.

Jika Anda menghapus salinan snapshot saat operasi penyalinan snapshot berbasis waktu masih berlangsung, Anda akan ditagih untuk data yang disalin hingga titik tersebut dengan tarif yang sesuai dengan durasi penyelesaian yang ditentukan.

Enkripsi dan penyalinan snapshot

> **Note**
>
> Enkripsi sisi server Amazon S3 (256-bit AES) melindungi data bergerak snapshot selama operasi penyalinan.

Anda dapat membuat salinan snapshot terenkripsi dari snapshot sumber yang tidak dienkripsi. Dan Anda dapat mengenkripsi salinan snapshot dengan kunci KMS yang berbeda dari snapshot sumber. Namun, mengubah status enkripsi salinan snapshot selama operasi penyalinan dapat menghasilkan salinan penuh (bukan tambahan), yang mungkin dikenakan biaya transfer data dan penyimpanan yang lebih besar.

> **Tip**
>
> Saat menggunakan snapshot terenkripsi yang dibagikan dengan Anda, sebaiknya Anda mengenkripsi ulang snapshot dengan menyalinnya dan menggunakan kunci KMS yang Anda miliki. Ini melindungi Anda jika kunci KMS asli dikompromikan, atau jika pemilik mencabut akses Anda, yang dapat menyebabkan Anda kehilangan akses ke snapshot dan volume terenkripsi apa pun yang Anda buat darinya.

Izin untuk menyalin snapshot terenkripsi

Untuk menyalin snapshot terenkripsi, pengguna Anda harus memiliki izin berikut untuk menggunakan enkripsi Amazon EBS.

- • `kms:DescribeKey`

- `kms:CreateGrant`
- `kms:GenerateDataKey`
- `kms:GenerateDataKeyWithoutPlaintext`
- `kms:ReEncrypt`
- `kms:Decrypt`

- Untuk menyalin snapshot terenkripsi yang dibagikan dari AWS akun lain, Anda harus memiliki izin untuk menggunakan kunci yang dikelola pelanggan yang digunakan untuk mengenkripsi snapshot tersebut. Untuk informasi selengkapnya, lihat Bagikan kunci KMS yang digunakan untuk mengenkripsi snapshot Amazon EBS bersama.

Hasil enkripsi untuk salinan snapshot

Tabel berikut menjelaskan hasil enkripsi saat menyalin snapshot yang Anda miliki dan snapshot yang dibagikan dengan Anda.

Enkripsi secara default untuk Wilayah tujuan	Snapshot sumber	Hasil enkripsi salinan snapshot	Catatan
Nonaktif	Tidak terenkripsi	Enkripsi opsional	Jika Anda mengenkripsi salinan, Anda dapat menentukan kunci KMS yang akan digunakan . Jika Anda mengenkripsi salinan tetapi tidak menentukan kunci KMS, kunci yang ditentukan untuk enkripsi secara default digunakan.
Nonaktif	Dienkripsi	Secara otomatis dienkripsi	Anda dapat menentukan kunci KMS yang akan digunakan. Jika Anda tidak menentukan kunci KMS, Kunci yang dikelola AWS (`aws/ebs`) digunakan.
Diaktifkan	Tidak terenkripsi	Secara otomatis dienkripsi	Anda dapat menentukan kunci KMS yang akan digunakan. Jika Anda tidak menentukan kunci KMS, kunci yang ditentukan untuk enkripsi secara default digunakan.

Enkripsi secara default untuk Wilayah tujuan	Snapshot sumber	Hasil enkripsi salinan snapshot	Catatan
Diaktifkan	Dienkripsi	Secara otomatis dienkripsi	Anda dapat menentukan kunci KMS yang akan digunakan. Jika Anda tidak menentukan kunci KMS, kunci yang ditentukan untuk enkripsi secara default digunakan.

Menyalin snapshot

Anda dapat menyalin snapshot dari satu Wilayah ke Wilayah lainnya. Anda dapat menyalin snapshot yang tidak dienkripsi ke snapshot terenkripsi. Namun, jika Anda mencoba menyalin snapshot terenkripsi tanpa memiliki izin untuk menggunakan kunci enkripsi, operasi gagal secara diam-diam dan salinan snapshot menerima pesan status “ID kunci yang diberikan tidak dapat diakses”.

Console

Untuk menyalin snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot yang akan dihapus, lalu pilih Tindakan, Salin snapshot.
4. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat untuk salinan snapshot tersebut.

   Secara default, deskripsi mencakup informasi tentang snapshot sumber sehingga Anda dapat mengidentifikasi salinan dari versi asli.
5. Tentukan tujuan untuk salinan snapshot.

   - Untuk menyalin snapshot ke Wilayah yang sama atau ke Wilayah yang berbeda, pilih AWS Wilayah dan kemudian pilih Wilayah tujuan.
   - Untuk menyalin snapshot ke Zona Lokal, pilih Zona AWS Lokal dan kemudian pilih Zona Lokal tujuan.
   - (Outpostkhusus pelanggan) Untuk menyalin snapshot ke sebuahOutpost, pilih AWS Outpost dan kemudian masukkan ARN tujuanOutpost.

6. Jika Anda ingin salinan snapshot diselesaikan dalam jangka waktu tertentu, pilih Aktifkan salinan berbasis waktu. Untuk Durasi penyelesaian, masukkan durasi penyelesaian yang diperlukan, dengan penambahan 15 menit. Untuk informasi selengkapnya, lihat Time-based salinan untuk snapshot dan AMI Amazon EBS EBS-backed.

   Jika Anda tidak memerlukan salinan snapshot untuk diselesaikan dalam jangka waktu tertentu, jangan aktifkan salinan berbasis waktu. Dalam hal ini, salinan snapshot diselesaikan berdasarkan upaya terbaik.
7. (Outpostkhusus pelanggan) Untuk membuat salinan snapshot di Wilayah yang dipilih, untuk tujuan Snapshot pilih AWS Outpost, dan kemudian untuk Outpost ARN Tujuan, masukkan ARN tempat Outpost untuk menyalin snapshot. Outpost Bid ang tujuan Snapshot hanya muncul jika Anda memiliki dan Outpost di Wilayah yang dipilih.
8. Tentukan status enkripsi untuk salinan snapshot.

   Jika snapshot sumber dienkripsi, atau jika akun Anda diaktifkan untuk enkripsi secara default, salinan snapshot akan dienkripsi secara otomatis. Jika snapshot sumber tidak dienkripsi dan akun Anda tidak diaktifkan untuk enkripsi secara default, enkripsi bersifat opsional.
9. Pilih Salin snapshot.

AWS CLI

Untuk menyalin snapshot ke Wilayah lain

Gunakan perintah copy-snapshot. Contoh berikut menyalin snapshot yang ditentukan dari Wilayah sumber ke Wilayah saat ini, yang ditentukan oleh `--region` opsi.

```
aws ec2 copy-snapshot \
    --source-snapshot-id snap-0abcdef1234567890 \
    --source-region us-east-1 \
    --region us-west-2
```

Untuk menyalin snapshot yang tidak dienkripsi ke snapshot terenkripsi

Gunakan perintah copy-snapshot. Contoh berikut menyalin snapshot tidak terenkripsi yang ditentukan dari Wilayah sumber ke Wilayah saat ini, mengenkripsi snapshot baru menggunakan kunci KMS yang ditentukan.

```
aws ec2 copy-snapshot \
```

```
    --source-snapshot-id snap-0abcdef1234567890 \
    --source-region us-east-1 \
    --encrypted \
    --kms-key-id alias/my-kms-key
```

PowerShell

Untuk menyalin snapshot ke Wilayah lain

Gunakan Copy-EC2Snapshot cmdlet. Contoh berikut menyalin snapshot yang ditentukan dari Wilayah sumber ke Wilayah saat ini, yang ditentukan oleh `--region` opsi.

```
Copy-EC2Snapshot `
    -SourceSnapshotId snap-0abcdef1234567890 `
    -SourceRegion us-east-1 `
    -Region us-west-2
```

Untuk menyalin snapshot yang tidak dienkripsi ke snapshot terenkripsi

Gunakan Copy-EC2Snapshot cmdlet. Contoh berikut menyalin snapshot tidak terenkripsi yang ditentukan dari Wilayah sumber ke Wilayah saat ini, mengenkripsi snapshot baru menggunakan kunci KMS yang ditentukan.

```
Copy-EC2Snapshot `
    -SourceSnapshotId snap-0abcdef1234567890 `
    -SourceRegion us-east-1 `
    -Encrypted $true `
    -KmsKeyId alias/my-kms-key
```

Bagikan snapshot Amazon EBS dengan orang lain AWS rekening

Anda dapat memodifikasi izin dari snapshot jika Anda ingin berbagi dengan akun AWS lainnya. Anda dapat membagikan snapshot secara publik dengan semua AWS akun lain, atau Anda dapat membagikannya secara pribadi dengan AWS akun individual yang Anda tentukan. Pengguna yang Anda beri wewenang dapat menggunakan snapshot yang Anda bagikan untuk membuat volume EBS mereka sendiri, sementara snapshot asli Anda tetap tidak terpengaruh.

> **Important**
>
> Saat Anda berbagi snapshot, Anda memberikan akses ke semua data di snapshot kepada orang lain. Bagikan snapshot hanya kepada individu yang Anda percayai dengan semua data snapshot Anda.

Untuk mencegah berbagi foto secara publik, Anda dapat mengaktifkanBlokir akses publik untuk snapshot Amazon EBS.

Topik

- Sebelum Anda berbagi snapshot
- Membagikan snapshot
- Bagikan kunci KMS yang digunakan untuk mengenkripsi snapshot Amazon EBS bersama
- Gunakan snapshot Amazon EBS yang dibagikan dengan Anda
- Menentukan penggunaan snapshot yang Anda bagikan

Sebelum Anda berbagi snapshot

Hal-hal berikut berlaku saat berbagi snapshot:

- Jika pemblokiran akses publik untuk snapshot diaktifkan untuk Wilayah, upaya guna membagikan snapshot secara publik akan diblokir. Snapshot masih dapat dibagikan secara privat.
- Snapshot dibatasi untuk Wilayah tempatnya dibuat. Untuk berbagi snapshot dengan Wilayah lain, salin snapshot ke Wilayah tersebut, lalu bagikan salinannya. Untuk informasi selengkapnya, lihat Menyalin snapshot Amazon EBS.
- Anda tidak dapat berbagi snapshot yang dienkripsi dengan Kunci yang dikelola AWS default. Anda hanya dapat berbagi snapshot yang dienkripsi dengan kunci yang dikelola pelanggan. Untuk informasi selengkapnya, lihat Membuat Kunci di Panduan Developer AWS Key Management Service .
- Anda hanya dapat berbagi snapshot yang tidak terenkripsi secara publik.
- Saat Anda berbagi snapshot terenkripsi, Anda juga harus berbagi kunci yang dikelola pelanggan yang digunakan untuk mengenkripsi snapshot. Untuk informasi selengkapnya, lihat Bagikan kunci KMS yang digunakan untuk mengenkripsi snapshot Amazon EBS bersama.

Membagikan snapshot

Anda dapat membagikan snapshot secara publik atau dengan AWS akun tertentu.

Console

Untuk berbagi snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot yang akan dibagikan, lalu pilih Tindakan, Ubah izin.
4. Tentukan izin snapshot. Pengaturan saat ini menunjukkan izin berbagi snapshot saat ini.
   - Untuk membagikan snapshot secara publik dengan semua AWS akun, pilih Pu blik.
   - Untuk membagikan snapshot secara pribadi dengan AWS akun tertentu, pilih Pri badi. Kemudian, di bagian Berbagi akun, pilih Tambah akun, dan masukkan 12 digit ID akun (tanpa tanda hubung) yang akan dibagikan.
5. Pilih Simpan perubahan.

AWS CLI

Izin untuk snapshot ditentukan menggunakan atribut `createVolumePermission` snapshot. Untuk membuat snapshot publik, atur grup ke `all`. Untuk berbagi snapshot dengan AWS akun tertentu, atur pengguna ke ID AWS akun.

Untuk berbagi snapshot secara publik

Gunakan perintah modify-snapshot-attribute.

Untuk `--attribute`, tentukan `createVolumePermission`. Untuk `--operation-type`, tentukan `add`. Untuk `--group-names`, tentukan `all`.

```
aws ec2 modify-snapshot-attribute \
    --snapshot-id snap-0abcdef1234567890 \
    --attribute createVolumePermission \
    --operation-type add \
    --group-names all
```

Untuk berbagi snapshot secara privat

Gunakan perintah modify-snapshot-attribute.

Untuk `--attribute`, tentukan `createVolumePermission`. Untuk `--operation-type`, tentukan `add`. Untuk`--user-ids`, tentukan ID 12 digit AWS akun yang digunakan untuk berbagi snapshot.

```
aws ec2 modify-snapshot-attribute \
    --snapshot-id snap-0abcdef1234567890 \
    --attribute createVolumePermission \
    --operation-type add \
    --user-ids 123456789012 111122223333
```

PowerShell

Izin untuk snapshot ditentukan menggunakan atribut `createVolumePermission` snapshot. Untuk membuat snapshot publik, atur grup ke `all`. Untuk berbagi snapshot dengan AWS akun tertentu, atur pengguna ke ID AWS akun.

Untuk berbagi snapshot secara publik

Gunakan Edit-EC2SnapshotAttribute cmdlet.

Untuk `-Attribute`, tentukan `CreateVolumePermission`. Untuk `-OperationType`, tentukan `Add`. Untuk `-GroupName`, tentukan `all`.

```
Edit-EC2SnapshotAttribute `
    -SnapshotId snap-0abcdef1234567890 `
    -Attribute CreateVolumePermission `
    -OperationType Add `
    -GroupName all
```

Untuk berbagi snapshot secara privat

Gunakan Edit-EC2SnapshotAttribute cmdlet.

Untuk `-Attribute`, tentukan `CreateVolumePermission`. Untuk `-OperationType`, tentukan `Add`. Untuk`UserId`, tentukan ID 12 digit AWS akun yang digunakan untuk berbagi snapshot.

```
Edit-EC2SnapshotAttribute `
    -SnapshotId snap-0abcdef1234567890 `
```

```
    -Attribute CreateVolumePermission `
    -OperationType Add `
    -UserId 123456789012 111122223333
```

Bagikan kunci KMS yang digunakan untuk mengenkripsi snapshot Amazon EBS bersama

Saat Anda berbagi snapshot terenkripsi, Anda juga harus berbagi kunci yang dikelola pelanggan yang digunakan untuk mengenkripsi snapshot. Anda dapat menerapkan izin lintas akun ke kunci yang dikelola pelanggan baik saat dibuat atau di lain waktu.

Pengguna kunci yang dikelola pelanggan bersama Anda yang mengakses snapshot terenkripsi harus diberikan izin untuk melakukan tindakan berikut pada kunci tersebut:

- `kms:DescribeKey`
- `kms:CreateGrant`
- `kms:GenerateDataKey`
- `kms:GenerateDataKeyWithoutPlaintext`
- `kms:ReEncrypt`
- `kms:Decrypt`

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan.

Untuk informasi selengkapnya tentang mengontrol akses ke kunci yang dikelola pelanggan, lihat Menggunakan kebijakan kunci di AWS KMS di Panduan Developer AWS Key Management Service .

Untuk membagikan kunci yang dikelola pelanggan menggunakan AWS KMS konsol

1. Buka AWS KMS konsol di https://console.aws.amazon.com/kms.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.

3. Pilih Kunci yang dikelola pelanggan di panel navigasi.
4. Di kolom Alias, pilih alias (tautan teks) dari kunci yang dikelola pelanggan yang Anda gunakan untuk mengenkripsi snapshot. Detail kunci terbuka di halaman baru.
5. Di bagian Kebijakan kunci, Anda melihat tampilan kebijakan atau tampilan default. Tampilan kebijakan menampilkan dokumen kebijakan kunci. Tampilan default menampilkan bagian untuk Administrator kunci, Penghapusan kunci, Penggunaan Kunci, dan Akun AWS lainnya. Tampilan default ditampilkan jika Anda membuat kebijakan di konsol dan belum menyesuaikannya. Jika tampilan default tidak tersedia, Anda perlu mengedit kebijakan secara manual dalam tampilan kebijakan. Untuk informasi selengkapnya, lihat Melihat Kebijakan Kunci (Konsol) dalam AWS Key Management Service Panduan Developer.

   Gunakan tampilan kebijakan atau tampilan default, tergantung pada tampilan mana yang dapat Anda akses, untuk menambahkan satu atau beberapa ID AWS akun ke kebijakan, sebagai berikut:

   - (Tampilan kebijakan) Pilih Edit. Tambahkan satu atau lebih ID AWS akun ke pernyataan berikut: `"Allow use of the key"` dan`"Allow attachment of persistent resources"`. Pilih Simpan perubahan. Dalam contoh berikut, ID AWS akun `444455556666` ditambahkan ke kebijakan.

     ```
     {
       "Sid": "Allow use of the key",
       "Effect": "Allow",
       "Principal": {"AWS": [
         "arn:aws:iam::111122223333:user/KeyUser",
         "arn:aws:iam::444455556666:root"
       ]},
       "Action": [
         "kms:Encrypt",
         "kms:Decrypt",
         "kms:ReEncrypt*",
         "kms:GenerateDataKey*",
         "kms:DescribeKey"
       ],
       "Resource": "*"
     },
     {
       "Sid": "Allow attachment of persistent resources",
       "Effect": "Allow",
       "Principal": {"AWS": [
         "arn:aws:iam::111122223333:user/KeyUser",
     ```

```
    "arn:aws:iam::444455556666:root"
  ]},
  "Action": [
    "kms:CreateGrant",
    "kms:ListGrants",
    "kms:RevokeGrant"
  ],
  "Resource": "*",
  "Condition": {"Bool": {"kms:GrantIsForAWSResource": true}}
}
```

- (Tampilan default) Gulir ke bawah ke AWS Akun lain. Pilih Tambahkan AWS akun lain dan masukkan ID AWS akun seperti yang diminta. Untuk menambahkan akun lain, pilih Tambahkan AWS akun lain dan masukkan ID AWS akun. Setelah Anda menambahkan semua akun AWS , pilih Simpan perubahan.

Gunakan snapshot Amazon EBS yang dibagikan dengan Anda

Untuk menggunakan snapshot yang dibagikan yang tidak dienkripsi

Cari snapshot yang dibagikan berdasarkan ID atau deskripsi. Anda dapat menggunakan snapshot ini sebagaimana dengan snapshot lain yang Anda miliki di akun Anda. Misalnya, Anda dapat membuat volume dari snapshot atau menyalinnya ke Wilayah yang berbeda.

Untuk menggunakan snapshot yang dibagikan yang terenkripsi

Cari snapshot yang dibagikan berdasarkan ID atau deskripsi. Buat salinan snapshot yang dibagikan di akun Anda, dan enkripsi salinannya dengan kunci KMS yang Anda miliki. Anda kemudian dapat menggunakan salinan untuk membuat volume atau Anda dapat menyalinnya ke Wilayah yang berbeda.

Console

Untuk melihat izin snapshot

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

4. Jika filter dimiliki oleh saya, snapshot dimiliki oleh akun ini. Jika filternya adalah Snapshot pribadi, snapshot dimiliki oleh akun ini atau dibagikan secara khusus dengan akun ini. Pilih snapshot dan pada tab Detail, periksa apakah Pemilik menentukan akun ini atau akun lain.

AWS CLI

Untuk melihat izin snapshot

Gunakan perintah describe-snapshot-atribut untuk mendapatkan izin snapshot dari snapshot yang ditentukan.

```
aws ec2 describe-snapshot-attribute \
    --snapshot-id snap-0abcdef1234567890 \
    --attribute createVolumePermission
```

Berikut ini adalah output contoh.

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "CreateVolumePermissions": [
        {
            "UserId": "111122223333"
        }
    ]
}
```

PowerShell

Untuk melihat izin snapshot

Gunakan Get-EC2SnapshotAttribute cmdlet.

```
(Get-EC2SnapshotAttribute `
    -SnapshotId snap-0abcdef1234567890 `
    -Attribute createVolumePermission).CreateVolumePermissions
```

Berikut ini adalah output contoh.

```
Group UserId
----- ------
      111122223333
```

Menentukan penggunaan snapshot yang Anda bagikan

Anda dapat menggunakannya AWS CloudTrail untuk memantau apakah snapshot yang telah Anda bagikan dengan orang lain disalin atau digunakan untuk membuat volume. Peristiwa berikut akan masuk CloudTrail saat tindakan diambil pada snapshot yang telah Anda bagikan:

- SharedSnapshotCopyInitiated — Snapshot yang sama sedang disalin.
- SharedSnapshotVolumeCreated — Foto bersama digunakan untuk membuat volume.

Untuk informasi selengkapnya tentang penggunaan CloudTrail, lihat Log panggilan Amazon EC2 dan Amazon EBS API dengan. AWS CloudTrail

Mengarsipkan snapshot Amazon EBS

Arsip Snapshot Amazon EBS adalah tingkat penyimpanan yang dapat Anda gunakan untuk penyimpanan berbiaya rendah dan jangka panjang dari snapshot yang jarang diakses yang tidak membutuhkan pengambilan yang sering dan cepat.

Secara default, saat Anda membuat snapshot, snapshot disimpan di tingkat Standar Snapshot Amazon EBS (tingkat standar). Snapshot yang disimpan di tingkat standar bersifat inkremental. Hal ini berarti bahwa hanya blok pada volume yang berubah setelah snapshot terbaru Anda disimpan.

Saat Anda mengarsipkan snapshot, snapshot inkremental akan dikonversi ke snapshot penuh, dan akan dipindahkan dari tingkat standar ke tingkat Arsip Snapshot Amazon EBS (tingkat arsip). Snapshot lengkap mencakup semua blok yang ditulis ke volume pada saat snapshot dibuat.

Saat Anda perlu mengakses snapshot yang diarsipkan, Anda dapat memulihkannya dari tingkat arsip ke tingkat standar, lalu menggunakannya dengan cara yang sama seperti Anda menggunakan snapshot lain di akun Anda.

Arsip Snapshot Amazon EBS menawarkan biaya penyimpanan snapshot hingga 75 persen lebih rendah untuk snapshot yang Anda rencanakan untuk disimpan selama 90 hari atau lebih lama dan yang jarang perlu Anda akses.

Beberapa kasus penggunaan khas meliputi:

- Mengarsipkan satu-satunya snapshot volume, seperti snapshot akhir proyek
- Mengarsipkan snapshot inkremental titik waktu penuh untuk alasan kepatuhan.
- Mengarsipkan snapshot inkremental bulanan, triwulanan, atau tahunan.

Topik

- Kuota
- Pertimbangan dan batasan untuk pengarsipan snapshot Amazon EBS
- Harga dan penagihan untuk pengarsipan snapshot Amazon EBS
- Pedoman dan praktik terbaik untuk mengarsipkan snapshot Amazon EBS
- Izin IAM yang diperlukan untuk mengarsipkan snapshot Amazon EBS
- Mengarsipkan snapshot Amazon EBS
- Memulihkan snapshot Amazon EBS yang diarsipkan
- Ubah periode pemulihan untuk snapshot Amazon EBS yang dipulihkan sementara
- Lihat snapshot Amazon EBS yang diarsipkan
- Memantau pengarsipan snapshot Amazon EBS menggunakan Acara CloudWatch

Kuota

Bagian ini menjelaskan kuota default untuk snapshot yang diarsipkan dan sedang berlangsung.

Kuota	Kuota default			
Snapshot yang diarsipkan per volume	25			
Arsip snapshot dalam proses bersamaan per akun	25			
Arsip snapshot dalam	5			

Kuota	Kuota default			
proses bersamaan per akun				

Jika Anda membutuhkan lebih dari batas default, lengkapi Dukungan formulir Center Cre ate case untuk meminta kenaikan batas.

Pertimbangan dan batasan untuk pengarsipan snapshot Amazon EBS

Ingatlah hal berikut saat mengarsipkan snapshot Amazon EBS.

Pertimbangan-pertimbangan

- Periode arsip minimum adalah 90 hari. Jika Anda menghapus atau memulihkan snapshot yang diarsipkan secara permanen sebelum periode arsip minimum 90 hari, Anda akan ditagih untuk sisa hari di tingkat arsip, dibulatkan ke jam terdekat. Untuk informasi selengkapnya, lihat Harga dan penagihan untuk pengarsipan snapshot Amazon EBS.
- Diperlukan waktu hingga 72 jam untuk memulihkan snapshot yang diarsipkan dari tingkat arsip ke tingkat standar, tergantung pada ukuran snapshot.
- Snapshot yang diarsipkan selalu merupakan snapshot penuh. Snapshot lengkap berisi semua blok yang ditulis ke volume pada saat snapshot dibuat. Snapshot lengkap kemungkinan akan lebih besar dari snapshot inkremental dari tempat snapshot tersebut dibuat. Namun, jika Anda hanya memiliki satu snapshot volume pada tingkat standar, ukuran snapshot penuh di tingkat arsip akan sama dengan ukuran snapshot di tingkat standar. Ini karena snapshot pertama yang diambil dari sebuah volume selalu merupakan snapshot penuh. Untuk mendapatkan ukuran snapshot penuh, gunakan perintah describe-snapshots AWS CLI .
- Pengarsipan direkomendasikan untuk snapshot bulanan, triwulanan, atau tahunan. Mengarsipkan snapshot inkremental harian dari satu volume dapat menyebabkan biaya yang lebih tinggi jika dibandingkan dengan menyimpannya di tingkat standar.
- Saat snapshot diarsipkan, data snapshot yang direferensikan oleh snapshot lain dalam garis keturunan snapshot dipertahankan di tingkat standar. Biaya data dan penyimpanan yang terkait dengan data yang direferensikan yang disimpan pada tingkat standar dialokasikan ke snapshot berikutnya dalam garis keturunan. Ini memastikan bahwa snapshot berikutnya dalam garis keturunan tidak terpengaruh oleh arsip.

- Jika Anda menghapus snapshot yang diarsipkan yang cocok dengan aturan retensi Keranjang Sampah, snapshot yang diarsipkan akan disimpan di Keranjang Sampah untuk periode retensi yang ditentukan dalam aturan retensi. Untuk menggunakan snapshot, Anda harus terlebih dahulu memulihkannya dari Keranjang Sampah, lalu mengembalikannya dari tingkat arsip. Untuk informasi selengkapnya, lihat Recycle Bin dan Harga dan penagihan untuk pengarsipan snapshot Amazon EBS.
- Anda tidak dapat menggunakan snapshot yang diarsipkan dalam pemetaan perangkat blok atau untuk membuat volume Amazon EBS.
- Anda dapat mengarsipkan snapshot yang dibuat dengan AWS Backup menggunakan Konsol AWS Backup, API, atau alat baris perintah. Untuk informasi selengkapnya, lihat Membuat rencana cadangan di Panduan Developer AWS Backup .

Batasan

- Anda dapat mengarsipkan snapshot yang ada dalam status `completed` saja.
- Anda hanya dapat mengarsipkan snapshot yang Anda miliki di akun Anda. Untuk mengarsipkan snapshot yang dibagikan dengan Anda, pertama-tama salin snapshot ke akun Anda dan kemudian arsipkan salinan snapshot.
- Sebelum dapat menggunakan snapshot yang diarsipkan, Anda harus terlebih dahulu memulihkannya ke tingkat standar. Memulihkan ke tingkat standar diperlukan untuk membuat volume dari snapshot melalui operasi API `CreateVolume` dan `RunInstances` serta untuk berbagi atau menyalin snapshot. Untuk informasi selengkapnya, lihat Memulihkan snapshot Amazon EBS yang diarsipkan.
- Anda dapat mengarsipkan snapshot yang dikaitkan dengan satu atau beberapa AMI hanya jika semua AMI terkait dinonaktifkan. Untuk informasi selengkapnya, lihat Menonaktifkan AMI.
- Anda tidak dapat mengaktifkan AMI yang dinonaktifkan jika snapshot terkait dipulihkan sementara. Semua snapshot terkait harus dipulihkan secara permanen sebelum Anda dapat mengaktifkan AMI.
- Anda tidak dapat membatalkan arsip snapshot atau proses pemulihan snapshot setelah dimulai.
- Anda dapat mengunci snapshot yang diarsipkan. Jika Anda mengarsipkan snapshot yang telah Anda bagikan dengan akun lain, akun yang digunakan untuk berbagi snapshot kehilangan akses setelah snapshot diarsipkan.
- Anda dapat menyalin snapshot yang diarsipkan. Jika Anda perlu menyalin snapshot yang diarsipkan, Anda harus memulihkannya terlebih dahulu.

- Anda tidak dapat mengaktifkan pemulihan snapshot cepat untuk snapshot yang diarsipkan. Pemulihan snapshot cepat dinonaktifkan secara otomatis saat snapshot diarsipkan. Jika Anda perlu menggunakan pemulihan snapshot cepat, Anda harus mengaktifkannya secara manual setelah memulihkan snapshot.

Harga dan penagihan untuk pengarsipan snapshot Amazon EBS

Snapshot yang diarsipkan ditagih dengan tarif $0,0125 per. GB-month Misalnya, jika Anda mengarsipkan snapshot sebesar 100 GiB, Anda dikenai biaya sebesar 1,25 USD (100 GiB * 0,0125 USD) per bulan.

Pemulihan snapshot dikenai biaya dengan tarif 0,03 USD per GB data yang dipulihkan. Misalnya, jika Anda mengembalikan snapshot 100 GiB dari tingkat arsip, Anda akan dikenai biaya satu kali sebesar 3 USD (100 GiB * 0,03 USD).

Setelah snapshot dikembalikan ke tingkat standar, snapshot akan ditagih dengan tarif standar untuk snapshot sebesar $0,05 per. GB-month

Untuk informasi selengkapnya, lihat harga Amazon EBS.

Penagihan untuk periode arsip minimum

Periode arsip minimum adalah 90 hari. Jika Anda menghapus atau memulihkan snapshot yang diarsipkan secara permanen sebelum periode arsip minimum 90 hari, Anda akan ditagih biaya prorata yang sama dengan biaya penyimpanan tingkat arsip untuk sisa harinya, dibulatkan ke jam terdekat. Misalnya, jika Anda menghapus atau memulihkan snapshot yang diarsipkan secara permanen setelah 40 hari, Anda akan dikenai biaya selama 50 hari tersisa dari periode arsip minimum.

> **Note**
>
> Memulihkan snapshot yang diarsipkan sementara sebelum periode arsip minimum 90 hari tidak dikenai biaya ini.

Pemulihan sementara

Saat Anda memulihkan snapshot sementara, snapshot dipulihkan dari tingkat arsip ke tingkat standar, dan salinan snapshot tetap berada di tingkat arsip. Anda dikenai biaya untuk snapshot di

tingkat standar dan salinan snapshot di tingkat arsip selama periode pemulihan sementara. Ketika snapshot yang dipulihkan sementara dihapus dari tingkat standar, Anda tidak lagi ditandaih untuk itu, dan Anda ditandaih untuk snapshot di tingkat arsip saja.

Pemulihan permanen

Saat Anda memulihkan snapshot sementara, snapshot dipulihkan dari tingkat arsip ke tingkat standar, dan salinan snapshot tetap berada di tingkat arsip. Anda ditandaih untuk snapshot di tingkat standar saja.

Menghapus snapshot

Jika Anda menghapus snapshot saat sedang diarsipkan, Anda akan ditandaih untuk data snapshot yang telah dipindahkan ke tingkat arsip. Data ini tunduk pada periode arsip minimum 90 hari dan ditandaih sesuai pada saat penghapusan. Misalnya, jika Anda mengarsipkan snapshot 100 GiB, dan menghapus snapshot setelah hanya 40 GiB diarsipkan, Anda akan ditagih $1,50 untuk periode arsip minimum 90 hari untuk 40 GiB yang telah diarsipkan ($0,0125 per GB-month * 40 GB * (90 hari * 24 jam)/(24 * 30 hari bulan). hours/day

Jika Anda menghapus snapshot saat sedang dipulihkan dari tingkat arsip, Anda akan dikenai biaya untuk pemulihan snapshot untuk ukuran penuh snapshot (ukuran snapshot * $0,03). Misalnya, jika Anda mengembalikan snapshot 100 GiB dari tingkat arsip, dan Anda menghapus snapshot kapan saja sebelum pemulihan snapshot selesai, Anda ditandaih $3 (ukuran snapshot 100 GiB* $0,03).

Keranjang Sampah

Snapshot yang diarsipkan dikenai biaya dengan tarif untuk snapshot yang diarsipkan saat berada di Keranjang Sampah. Snapshot yang diarsipkan yang ada di Keranjang Sampah tunduk pada periode arsip minimum 90 hari dan dikenai biaya sesuai jika dihapus oleh Keranjang Sampah sebelum periode arsip minimum. Dengan kata lain, jika aturan retensi menghapus snapshot yang diarsipkan dari Keranjang Sampah sebelum periode minimum 90 hari, Anda akan dikenai biaya untuk sisa hari.

Jika Anda menghapus snapshot yang cocok dengan aturan retensi saat snapshot sedang diarsipkan, snapshot yang diarsipkan akan disimpan di Keranjang Sampah untuk periode retensi yang ditentukan dalam aturan retensi. Itu dikenai biaya dengan tarif untuk snapshot yang diarsipkan.

Jika Anda menghapus snapshot yang cocok dengan aturan retensi saat snapshot dipulihkan, snapshot yang dipulihkan akan disimpan di Keranjang Sampah selama sisa periode retensi, dan dikenai biaya pada tingkat snapshot standar. Untuk menggunakan snapshot yang dipulihkan, Anda harus memulihkannya terlebih dahulu dari Keranjang Sampah.

Untuk informasi selengkapnya, lihat Recycle Bin.

Pelacakan biaya

Snapshot yang diarsipkan muncul AWS Cost and Usage Report dengan ID sumber daya dan Nama Sumber Daya Amazon (ARN) yang sama. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Cost and Usage Report.

Anda dapat menggunakan tipe penggunaan berikut untuk mengidentifikasi biaya terkait:

- `SnapshotArchiveStorage` — biaya untuk penyimpanan data bulanan
- `SnapshotArchiveRetrieval` — biaya satu kali untuk pemulihan snapshot
- `SnapshotArchiveEarlyDelete` — biaya untuk menghapus atau memulihkan snapshot secara permanen sebelum periode arsip minimum (90 hari)

Pedoman dan praktik terbaik untuk mengarsipkan snapshot Amazon EBS

Bagian ini memberikan beberapa pedoman dan praktik terbaik untuk mengarsipkan snapshot.

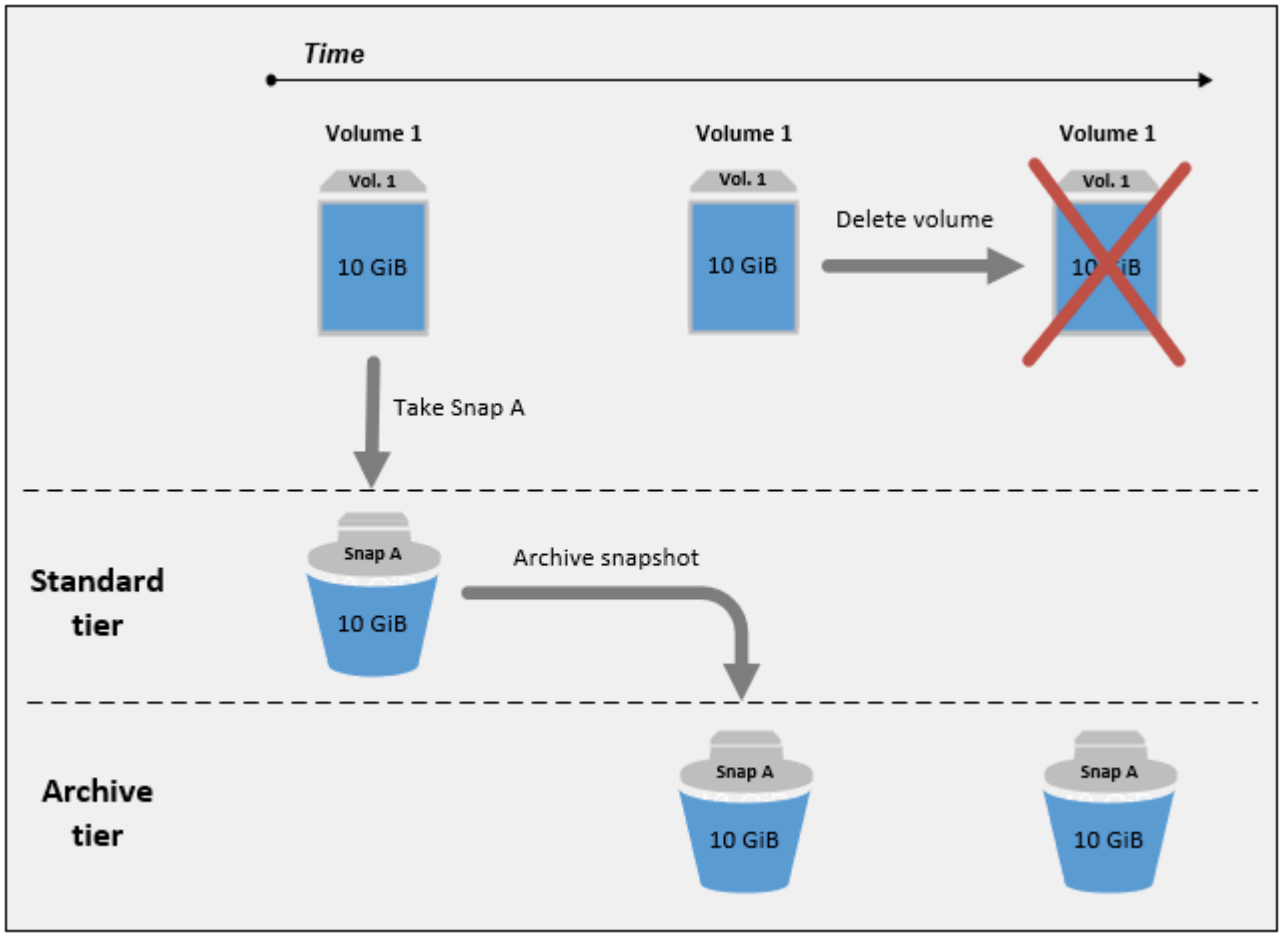
Topik

- Mengarsipkan satu-satunya snapshot volume
- Mengarsipkan snapshot tambahan dari satu volume
- Mengarsipkan snapshot lengkap untuk alasan kepatuhan
- Menentukan pengurangan biaya penyimpanan tingkat standar

Mengarsipkan satu-satunya snapshot volume

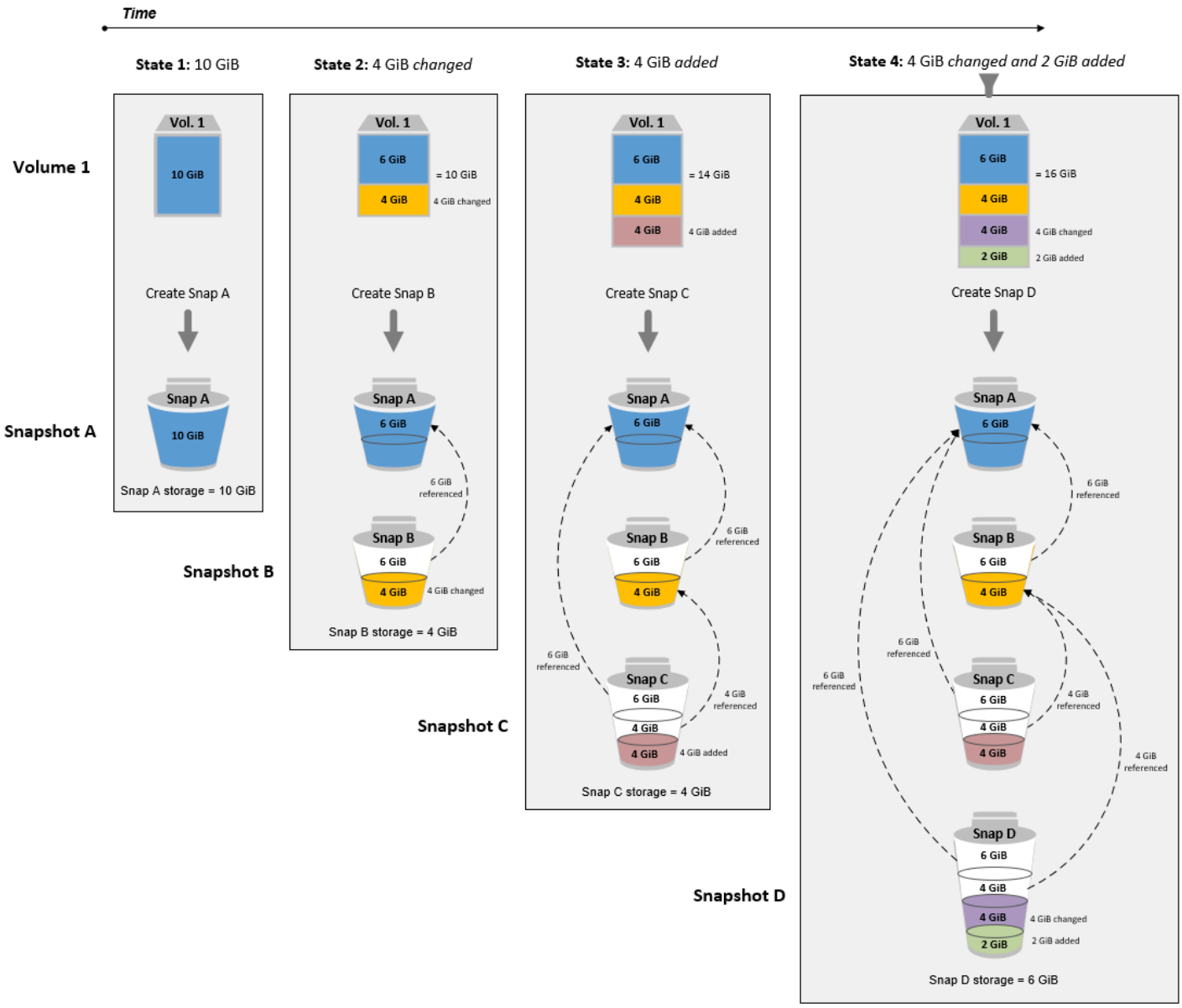
Bila Anda hanya memiliki satu snapshot volume, snapshot selalu berukuran sama dengan blok yang ditulis ke volume pada saat snapshot dibuat. Saat Anda mengarsipkan snapshot seperti itu, snapshot di tingkat standar dikonversi ke snapshot penuh berukuran setara dan dipindahkan dari tingkat standar ke tingkat arsip.

Mengarsipkan snapshot ini dapat membantu Anda menghemat dengan biaya penyimpanan yang lebih rendah. Jika Anda tidak lagi membutuhkan volume sumber, Anda dapat menghapus volume untuk penghematan biaya penyimpanan lebih lanjut.



Mengarsipkan snapshot tambahan dari satu volume

Saat Anda mengarsipkan snapshot inkremental, snapshot akan dikonversi ke snapshot penuh dan akan dipindahkan ke tingkat Arsip. Misalnya, pada gambar berikut, jika Anda mengarsipkan Snap B, snapshot dikonversi ke snapshot penuh yang berukuran 10 GiB dan dipindahkan ke tingkat arsip. Demikian pula, jika Anda mengarsipkan Snap C, ukuran snapshot lengkap di tingkat arsip adalah 14 GiB.



Jika Anda mengarsipkan snapshot untuk mengurangi biaya penyimpanan di tingkat standar, Anda tidak boleh mengarsipkan snapshot pertama dalam satu set snapshot tambahan. Snapshot ini direferensikan oleh snapshot berikutnya dalam garis keturunan snapshot. Dalam kebanyakan kasus, pengarsipan snapshot ini tidak akan mengurangi biaya penyimpanan.

> **Note**
>
> Anda tidak boleh mengarsipkan snapshot terakhir dalam satu set snapshot tambahan. Snapshot terakhir adalah snapshot terbaru yang diambil dari sebuah volume. Anda akan

memerlukan snapshot ini di tingkat standar jika Anda ingin membuat volume darinya jika terjadi korupsi atau kehilangan volume.

Jika Anda mengarsipkan snapshot yang berisi data yang direferensikan oleh snapshot berikutnya di garis keturunan, biaya penyimpanan dan penyimpanan data yang terkait dengan data yang direferensikan dialokasikan ke snapshot berikutnya dalam garis keturunan. Dalam kasus ini, pengarsipan snapshot tidak akan mengurangi biaya penyimpanan atau penyimpanan data. Misalnya, pada gambar sebelumnya, jika Anda mengarsipkan Snap B, 4 GiB datanya diatribusikan ke Snap C. Dalam hal ini, biaya penyimpanan Anda secara keseluruhan akan meningkat karena Anda dikenai biaya penyimpanan untuk versi lengkap Snap B di tingkat arsip, dan biaya penyimpanan Anda untuk tingkat standar tetap tidak berubah.

Jika Anda mengarsipkan Snap C, penyimpanan tingkat standar Anda akan berkurang sebesar 4 GiB karena data tidak direferensikan oleh snapshot lain nanti di garis keturunan. Dan penyimpanan tingkat arsip Anda akan meningkat sebesar 14 GiB karena snapshot dikonversi ke snapshot penuh.

Mengarsipkan snapshot lengkap untuk alasan kepatuhan

Anda mungkin perlu membuat cadangan volume penuh setiap bulan, triwulanan, atau tahunan untuk alasan kepatuhan. Untuk pencadangan ini, Anda mungkin memerlukan snapshot mandiri tanpa referensi mundur atau meneruskan ke snapshot lain di garis keturunan snapshot. Snapshot yang diarsipkan dengan EBS Snapshots Archive adalah snapshot lengkap, dan tidak memiliki referensi ke snapshot lain dalam garis keturunan. Selain itu, Anda mungkin perlu mempertahankan snapshot ini untuk alasan kepatuhan selama beberapa tahun. EBS Snapshots Archive membuatnya hemat biaya untuk mengarsipkan snapshot lengkap ini untuk retensi jangka panjang.

Menentukan pengurangan biaya penyimpanan tingkat standar

Jika Anda ingin mengarsipkan snapshot tambahan untuk mengurangi biaya penyimpanan Anda, Anda harus mempertimbangkan ukuran snapshot penuh di tingkat arsip dan pengurangan penyimpanan di tingkat standar. Bagian ini menjelaskan cara melakukannya.

Important

Respons API adalah data yang akurat pada saat API dipanggil. Respons API dapat berbeda karena data yang terkait dengan snapshot berubah sebagai akibat dari perubahan garis keturunan snapshot.

Untuk menentukan pengurangan biaya penyimpanan dan penyimpanan di tingkat standar, gunakan langkah-langkah berikut.

1. Untuk snapshot yang ingin Anda arsipkan, periksa ukuran snapshot penuh dan volume sumber dari mana ia dibuat. Gunakan perintah describe-snapshots, dan untuk`--snapshot-id`, tentukan ID snapshot yang ingin Anda arsipkan.

   ```
   $ aws ec2 describe-snapshots --snapshot-id snapshot_id
   ```

   Nilai `FullSnapshotSizeInBytes` reponse menunjukkan ukuran snapshot penuh, dalam byte, dan nilai `VolumeId` respons menunjukkan ID volume sumber.

   Misalnya, perintah berikut memberikan informasi tentang snapshot `snap-09c9114207084f0d9`.

   ```
   $ aws ec2 describe-snapshots --snapshot-id snap-09c9114207084f0d9
   ```

   Contoh output berikut menunjukkan bahwa ukuran snapshot penuh adalah `5678912341` byte (5,28 GiB), dan volume sumbernya adalah. `vol-0f3e2c292c52b85c3`

   ```
   {
       "Snapshots": [
           {
               "Description": "",
               "Tags": [],
               "Encrypted": false,
               "VolumeId": "vol-0f3e2c292c52b85c3",
               "State": "completed",
               "VolumeSize": 8,
               "StartTime": "2021-11-16T08:29:49.840Z",
               "Progress": "100%",
               "OwnerId": "123456789012",
               "FullSnapshotSizeInBytes" : "5678912341",
               "SnapshotId": "snap-09c9114207084f0d9"
           }
       ]
   }
   ```

2. Temukan semua snapshot yang dibuat dari volume sumber. Gunakan perintah describe-snapshots. Tentukan filter `volume-id`, dan untuk nilai filter, tentukan ID volume dari langkah sebelumnya.

```
$ aws ec2 describe-snapshots --filters "Name=volume-id, Values=volume_id"
```

Misalnya, perintah berikut mengembalikan semua snapshot yang dibuat dari volume `vol-0f3e2c292c52b85c3`.

```
$ aws ec2 describe-snapshots --filters "Name=volume-id,
 Values=vol-0f3e2c292c52b85c3"
```

Berikut ini adalah output perintah, yang menunjukkan bahwa tiga snapshot dibuat dari volume `vol-0f3e2c292c52b85c3`.


```
{
    "Snapshots": [
        {
            "Description": "",
            "Tags": [],
            "Encrypted": false,
            "VolumeId": "vol-0f3e2c292c52b85c3",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2021-11-14T08:57:39.300Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-08ca60083f86816b0"
        },
        {
            "Description": "",
            "Tags": [],
            "Encrypted": false,
            "VolumeId": "vol-0f3e2c292c52b85c3",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2021-11-15T08:29:49.840Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-09c9114207084f0d9"
        },
```

```
        {
            "Description": "01",
            "Tags": [],
            "Encrypted": false,
            "VolumeId": "vol-0f3e2c292c52b85c3",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2021-11-16T07:50:08.042Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-024f49fe8dd853fa8"
        }
    ]
}
```

3. Menggunakan output dari perintah sebelumnya, urutkan snapshot berdasarkan waktu pembuatannya, dari yang tertua ke yang terbaru. Parameter `StartTime` respons untuk setiap snapshot menunjukkan waktu pembuatannya, dalam format waktu UTC.

   Misalnya, snapshot yang dikembalikan pada langkah sebelumnya yang diatur berdasarkan waktu pembuatan, dari tertua ke terbaru, adalah sebagai berikut:

   1. `snap-08ca60083f86816b0`(tertua - dibuat sebelum snapshot yang ingin Anda arsipkan)
   2. `snap-09c9114207084f0d9` (snapshot untuk diarsipkan)
   3. `snap-024f49fe8dd853fa8` (terbaru — dibuat setelah snapshot yang ingin Anda arsipkan)

4. Identifikasi snapshot yang dibuat segera sebelum dan sesudah snapshot yang ingin Anda arsipkan. Dalam hal ini, Anda harus mengarsipkan snapshot `snap-09c9114207084f0d9`, yang merupakan snapshot inkrementa; kedua yang dibuat dalam rangkaian tiga snapshot. Snapshot `snap-08ca60083f86816b0` dibuat segera sebelumnya, dan snapshot `snap-024f49fe8dd853fa8` dibuat segera setelahnya.

5. Temukan data yang tidak direferensikan dalam snapshot yang ingin Anda arsipkan. Pertama, temukan blok yang berbeda antara snapshot yang dibuat segera sebelum snapshot yang ingin Anda arsipkan, dan snapshot yang ingin Anda arsipkan. Gunakan perintah list-changed-blocks. Untuk `--first-snapshot-id`, tentukan ID snapshot yang dibuat segera sebelum snapshot yang ingin Anda arsipkan. Untuk `--second-snapshot-id`, tentukan ID snapshot yang ingin Anda arsipkan.

   ```
   $ aws ebs list-changed-blocks --first-snapshot-id snapshot_created_before --second-
   snapshot-id snapshot_to_archive
   ```

Misalnya, perintah berikut menunjukkan indeks blok untuk blok yang berbeda antara snapshot `snap-08ca60083f86816b0` (snapshot yang dibuat sebelum snapshot yang ingin Anda arsipkan), dan snapshot `snap-09c9114207084f0d9` (snapshot yang ingin Anda arsipkan).

```
$ aws ebs list-changed-blocks --first-snapshot-id snap-08ca60083f86816b0 --second-
snapshot-id snap-09c9114207084f0d9
```

Berikut ini akan menunjukkan output perintah, dengan beberapa blok dihilangkan.

```
{
    "BlockSize": 524288,
    "ChangedBlocks": [
        {
            "FirstBlockToken": "ABgBAX6y
+WH6Rm9y5zq1VyeTCmEzGmTT0jNZG1cDirFq1rOVeFbWXsH3W4z/",
            "SecondBlockToken": "ABgBASyx0bHHBnTERu
+9USLxYK/81UT0dbHIUFqUjQUkwTwK5qkjP8NSGyNB",
            "BlockIndex": 4
        },
        {
            "FirstBlockToken": "ABgBAcfL
+EfmQmlNgstqrFnYgsAxR4SDSO4LkNLYOOChGBWcfJnpn90E9XX1",
            "SecondBlockToken": "ABgBAdX0mtX6aBAt3EBy
+8jFCESMpig7csKjbO2Ocd08m2iNJV2Ue+cRwUqF",
            "BlockIndex": 5
        },
        {
            "FirstBlockToken": "ABgBAVBaFJmbP/eRHGh7vnJlAwyiyNUi3MKZmEMxs2wC3AmM/
fc6yCOAMb65",
            "SecondBlockToken":
 "ABgBAdewWkHKTcrhZmsfM7GbaHyXD1Ctcn2nppz4wYItZRmAo1M72fpXU0Yv",
            "BlockIndex": 13
        },
        {
            "FirstBlockToken": "ABgBAQGxwuf6z095L6DpRoVRVnOqPxmx9r7Wf6O+i
+ltZ0dwPpGN39ijztLn",
            "SecondBlockToken": "ABgBAUdlitCVI7c6hGsT4ckkKCw6bMRclnV
+bKjViu/9UESTcW7CD9w4J2td",
            "BlockIndex": 14
        },
        {
```

```
            "FirstBlockToken":
 "ABgBAZBfEv4EHS1aSXTXxSE3mBZG6CNeIkwxpljzmgSHICGlFmZCyJXzE4r3",
            "SecondBlockToken":
 "ABgBAVWR7QuQQB0AP2TtmNkgS4Aec5KAQVCldnpc91zBiNmSfW9ouIlbeXWy",
            "BlockIndex": 15
        },
        .....
        {
            "SecondBlockToken": "ABgBAeHwXPL+z3DBLjDhwjdAM9+CPGV5VO5Q3rEEA
+ku50P498hjnTAgMhLG",
            "BlockIndex": 13171
        },
        {
            "SecondBlockToken":
 "ABgBAbZcPiVtLx6U3Fb4lAjRdrkJMwW5M2tiCgIp6ZZpcZ8AwXxkjVUUHADq",
            "BlockIndex": 13172
        },
        {
            "SecondBlockToken": "ABgBAVmEd/pQ9VW9hWiOujOAKcauOnUFCO
+eZ5ASVdWLXWWC04ijfoDTpTVZ",
            "BlockIndex": 13173
        },
        {
            "SecondBlockToken": "ABgBAT/jeN7w
+8ALuNdaiwXmsSfM6tOvMoLBLJ14LKvavw4IiB1d0iykWe6b",
            "BlockIndex": 13174
        },
        {
            "SecondBlockToken": "ABgBAXtGvUhTjjUqkwKXfXzyR2GpQei/
+pJSG/19ESwvt7Hd8GHaUqVs6Zf3",
            "BlockIndex": 13175
        }
    ],
    "ExpiryTime": 1637648751.813,
    "VolumeSize": 8
}
```

Selanjutnya, gunakan perintah yang sama untuk menemukan blok yang berbeda antara snapshot yang ingin Anda arsipkan dan snapshot yang dibuat segera setelahnya. Untuk `--first-snapshot-id`, tentukan ID snapshot yang ingin Anda arsipkan. Untuk `--second-`

`snapshot-id`, tentukan ID snapshot yang dibuat segera sebelum snapshot yang ingin Anda arsipkan.

```
$ aws ebs list-changed-blocks --first-snapshot-id snapshot_to_archive --second-
snapshot-id snapshot_created_after
```

Misalnya, perintah berikut menunjukkan indeks blok untuk blok yang berbeda antara snapshot `snap-09c9114207084f0d9` (snapshot yang dibuat sebelum snapshot yang ingin Anda arsipkan), dan snapshot `snap-024f49fe8dd853fa8` (snapshot yang ingin Anda arsipkan).

```
$ aws ebs list-changed-blocks --first-snapshot-id snap-09c9114207084f0d9 --second-
snapshot-id snap-024f49fe8dd853fa8
```

Berikut ini akan menunjukkan output perintah, dengan beberapa blok dihilangkan.

```
{
    "BlockSize": 524288,
    "ChangedBlocks": [
        {
            "FirstBlockToken": "ABgBAVax0bHHBnTERu
+9USLxYK/81UT0dbSnkDk0gqwRFSFGWA7HYbkkAy5Y",
            "SecondBlockToken":
 "ABgBASEvi9x8Om7Htp37cKG2NT9XUzEbLHpGcayelomSoHpGy8LGyvG0yYfK",
            "BlockIndex": 4
        },
        {
            "FirstBlockToken": "ABgBAeL0mtX6aBAt3EBy+8jFCESMpig7csfMrI4ufnQJT3XBm/
pwJZ1n2Uec",
            "SecondBlockToken": "ABgBAXmUTg6rAI
+v0LvekshbxCVpJjWILvxgC0AG0GQBEUNRVHkNABBwXLkO",
            "BlockIndex": 5
        },
        {
            "FirstBlockToken":
 "ABgBATKwWkHKTcrhZmsfM7GbaHyXD1CtcnjIZv9YzisYsQTMHfTfh4AhS0s2",
            "SecondBlockToken": "ABgBAcmiPFovWgXQio
+VBrxOqGy4PKZ9SAAHaZ2HQBM9fQQU0+EXxQjVGv37",
            "BlockIndex": 13
        },
        {
```

```
            "FirstBlockToken":
 "ABgBAbRlitCVI7c6hGsT4ckkKCw6bMRclnARrMt1hUbIhFnfz8kmUaZOP2ZE",
            "SecondBlockToken": "ABgBAXe935n544+rxhJ0INB8q7pAeoPZkkD27vkspE/
qKyvOwpozYII6UNCT",
            "BlockIndex": 14
        },
        {
            "FirstBlockToken": "ABgBAd+yxCO26I
+1Nm2KmuKfrhjCkuaP6LXuol3opCNk6+XRGcct4suBHje1",
            "SecondBlockToken": "ABgBAcPpnXz821NtTvWBPTz8uUFXnS8jXubvghEjZulIjHgc
+7saWys77shb",
            "BlockIndex": 18
        },
        .....
        {
            "SecondBlockToken": "ABgBATni4sDE5rS8/a9pqV03lU/lKCW
+CTxFl3cQ5p2f2h1njpuUiGbqKGUa",
            "BlockIndex": 13190
        },
        {
            "SecondBlockToken": "ABgBARbXo7zFhu7IEQ/9VMYFCTCtCuQ
+iSlWVpBIshmeyeS5FD/M0i64U+a9",
            "BlockIndex": 13191
        },
        {
            "SecondBlockToken": "ABgBAZ8DhMk+rROXa4dZlNK45rMYnVIGGSyTeiMli/sp/
JXUVZKJ9sMKIsGF",
            "BlockIndex": 13192
        },
        {
            "SecondBlockToken":
 "ABgBATh6MBVE904l6sqOC27s1nVntFUpDwiMcRWGyJHy8sIgGL5yuYXHAVty",
            "BlockIndex": 13193
        },
        {
            "SecondBlockToken":
 "ABgBARuZykaFBWpCWrJPXaPCneQMbyVgnITJqj4c1kJWPIj5Gn61OQyy+giN",
            "BlockIndex": 13194
        }
    ],
    "ExpiryTime": 1637692677.286,
    "VolumeSize": 8
}
```

6. Bandingkan output yang dikembalikan oleh kedua perintah pada langkah sebelumnya. Jika indeks blok yang sama muncul di kedua output perintah, ini menunjukkan bahwa blok berisi data yang tidak direferensikan.

   Misalnya, output perintah pada langkah sebelumnya menunjukkan bahwa blok 4, 5, 13, dan 14 unik untuk snapshot `snap-09c9114207084f0d9` dan bahwa mereka tidak direferensikan oleh snapshot lain dalam garis keturunan snapshot.

   Untuk menentukan pengurangan penyimpanan tingkat standar, kalikan jumlah blok yang muncul di kedua output perintah dengan 512 KiB, yang merupakan ukuran blok snapshot.

   Misalnya, jika 9.950 indeks blok muncul di kedua output perintah, ini menunjukkan bahwa Anda akan mengurangi penyimpanan tingkat standar sekitar 4,85 GiB (9.950 blok * 512 KiB = 4,85 GiB).
7. Tentukan biaya penyimpanan untuk menyimpan blok yang tidak direferensikan di tingkat standar selama 90 hari. Bandingkan nilai ini dengan biaya penyimpanan snapshot lengkap, dijelaskan pada langkah 1, di tingkat arsip. Anda dapat menentukan penghematan biaya dengan membandingkan nilainya, dengan asumsi bahwa Anda tidak memulihkan snapshot penuh dari tingkat arsip selama periode minimum 90 hari. Untuk informasi selengkapnya, lihat Harga dan penagihan untuk pengarsipan snapshot Amazon EBS.

Izin IAM yang diperlukan untuk mengarsipkan snapshot Amazon EBS

Secara default, pengguna tidak memiliki izin untuk menggunakan pengarsipan snapshot. Untuk mengizinkan pengguna bekerja dengan sumber daya ini, Anda harus membuat kebijakan IAM yang memberikan izin menggunakan sumber daya dan tindakan API tertentu. Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Untuk menggunakan pengarsipan snapshot, pengguna memerlukan izin berikut.

- `ec2:DescribeSnapshotTierStatus`
- `ec2:ModifySnapshotTier`
- `ec2:RestoreSnapshotTier`

Pengguna konsol mungkin memerlukan izin tambahan seperti `ec2:DescribeSnapshots`.

Untuk mengarsipkan dan memulihkan snapshot terenkripsi, AWS KMS izin tambahan berikut diperlukan.

- `kms:CreateGrant`
- `kms:Decrypt`
- `kms:DescribeKey`

Berikut ini adalah contoh kebijakan IAM yang memberikan izin kepada pengguna IAM untuk mengarsipkan, memulihkan, serta melihat snapshot terenkripsi dan tidak terenkripsi. Ini termasuk izin `ec2:DescribeSnapshots` untuk pengguna konsol. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "ec2:DescribeSnapshotTierStatus",
            "ec2:ModifySnapshotTier",
            "ec2:RestoreSnapshotTier",
            "ec2:DescribeSnapshots",
            "kms:CreateGrant",
            "kms:Decrypt",
            "kms:DescribeKey"
        ],
        "Resource": "*",
        "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": true
                }
            }
```

```
    }]
}
```

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Mengarsipkan snapshot Amazon EBS

Anda dapat mengarsipkan snapshot apa pun yang ada dalam status `completed` dan yang Anda miliki di akun Anda. Anda tidak dapat mengarsipkan snapshot yang ada di status `pending` atau `error`, atau snapshot yang dibagikan dengan Anda. Untuk informasi selengkapnya, lihat Pertimbangan dan batasan untuk pengarsipan snapshot Amazon EBS.

Jika snapshot dikaitkan dengan satu atau lebih AMI, Anda harus menonaktifkan AMI terkait terlebih dahulu sebelum Anda dapat mengarsipkan snapshot. Untuk informasi selengkapnya, lihat Menonaktifkan AMI.

Snapshot yang diarsipkan mempertahankan ID snapshot, status enkripsi, izin AWS Identity and Access Management (IAM), informasi pemilik, dan tag sumber daya. Namun, pemulihan snapshot cepat dan berbagi snapshot dinonaktifkan secara otomatis setelah snapshot diarsipkan.

Anda dapat terus menggunakan snapshot saat arsip sedang dalam proses. Segera setelah status tingkat snapshot mencapai status `archival-complete`, Anda tidak dapat lagi menggunakan snapshot.

Console

Untuk mengarsipkan snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Dalam daftar snapshot, pilih snapshot yang akan diarsipkan, kemudian pilih Tindakan, Arsipkan snapshot.
4. Untuk mengonfirmasi, pilih Pulihkan snapshot.

AWS CLI

Untuk mengarsipkan snapshot

Gunakan perintah modify- AWS CLI snapshot-tier. Untuk `--snapshot-id`, tentukan ID snapshot yang ingin Anda arsipkan. Untuk `--storage-tier`, tentukan `archive`.

```
aws ec2 modify-snapshot-tier \
    --snapshot-id snap-0abcdef1234567890 \
    --storage-tier archive
```

Berikut ini adalah output contoh. Parameter `TieringStartTime` respons menunjukkan tanggal dan waktu di mana proses arsip dimulai, dalam format waktu UTC (YYYY-MM-DDTHH:MM:SSZ).

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "TieringStartTime": "2021-09-15T16:44:37.574Z"
}
```

PowerShell

Untuk mengarsipkan snapshot

Gunakan Edit-EC2SnapshotTier cmdlet. Contoh berikut mengarsipkan snapshot yang ditentukan.

```
Edit-EC2SnapshotTier `
    -SnapshotId snap-0abcdef1234567890 `
    -StorageTier "archive"
```

Memulihkan snapshot Amazon EBS yang diarsipkan

Sebelum dapat menggunakan snapshot yang diarsipkan, Anda harus terlebih dahulu memulihkannya ke tingkat standar. Snapshot yang dipulihkan memiliki ID snapshot, status enkripsi, izin IAM, informasi pemilik, dan tanda sumber daya yang sama yang dimilikinya sebelum diarsipkan. Anda dapat menggunakan AMI yang dipulihkan dengan cara yang sama seperti Anda menggunakan AMI lainnya di akun Anda. Snapshot yang dipulihkan selalu merupakan snapshot penuh.

Saat memulihkan snapshot, Anda dapat memilih untuk memulihkannya secara permanen atau sementara.

Jika Anda memulihkan snapshot secara permanen, snapshot dipindahkan dari tingkat arsip ke tingkat standar secara permanen. Snapshot tetap dipulihkan dan siap digunakan sampai Anda mengarsipkan ulang secara manual atau Anda menghapusnya secara manual. Saat Anda memulihkan snapshot secara permanen, snapshot dihapus dari tingkat arsip.

Jika Anda memulihkan snapshot sementara, snapshot disalin dari tingkat arsip ke tingkat standar untuk periode pemulihan yang Anda tentukan. Snapshot tetap dipulihkan dan siap digunakan hanya untuk periode pemulihan. Selama periode pemulihan, salinan snapshot tetap berada di tingkat arsip. Setelah periode berakhir, snapshot secara otomatis dihapus dari tingkat standar. Anda dapat menambah atau mengurangi periode pemulihan atau mengubah tipe pemulihan menjadi permanen kapan saja selama periode pemulihan. Untuk informasi selengkapnya, lihat Ubah periode pemulihan untuk snapshot Amazon EBS yang dipulihkan sementara.

Jika Anda memulihkan snapshot yang terkait dengan AMI yang dinonaktifkan, dan Anda ingin menggunakan AMI tersebut, Anda harus terlebih dahulu memulihkan semua snapshot terkait secara permanen dan kemudian mengaktifkan kembali AMI yang dinonaktifkan sebelum Anda dapat menggunakannya. Anda tidak dapat mengaktifkan AMI jika snapshot terkait dipulihkan sementara. Anda dapat menggunakan perintah berikut untuk menemukan semua snapshot yang terkait dengan AMI.

```
aws ec2 describe-images --image-id ami_id \
  --query Images[*].BlockDeviceMappings[*].Ebs[].SnapshotId[]
```

Console

Untuk memulihkan snapshot dari arsip

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

1. Di panel navigasi, pilih Snapshot.
2. Dalam daftar snapshot, pilih snapshot yang akan diarsipkan, kemudian pilih Tindakan, Pulihkan snapshot dari arsip.
3. Tentukan jenis pemulihan yang akan dilakukan. Untuk Jenis pemulihan, lakukan salah satu langkah berikut:
   - Untuk memulihkan snapshot secara permanen, pilih Permanen.
   - Untuk memulihkan snapshot secara sementara, pilih Sementara, kemudian untuk Periode pemulihan sementara, masukkan jumlah hari untuk mengembalikan snapshot.
4. Untuk mengonfirmasi, pilih Pulihkan snapshot.

AWS CLI

Untuk memulihkan snapshot yang diarsipkan secara permanen

Gunakan perintah restore-snapshot- AWS CLI tier dengan opsi. `--permanent-restore` Untuk `--snapshot-id`, tentukan ID snapshot yang akan dipulihkan.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-0abcdef1234567890 \
    --permanent-restore
```

Berikut ini adalah output contoh.

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "IsPermanentRestore": true
}
```

Untuk memulihkan snapshot yang diarsipkan sementara

Gunakan perintah restore- AWS CLI snapshot-tier. Abaikan `--permanent-restore` opsi. Untuk`--temporary-restore-days`, tentukan jumlah hari untuk memulihkan snapshot. Kisaran yang diizinkan adalah 1 hingga 180 hari. Jika Anda tidak menentukan nilai, defaultnya adalah 1 hari.

Contoh berikut mengembalikan sementara snapshot yang ditentukan untuk periode pemulihan 5 hari.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-0abcdef1234567890 \
    --temporary-restore-days 5
```

Berikut ini adalah output contoh.

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "RestoreDuration": 5,
    "IsPermanentRestore": false
}
```

PowerShell

Untuk memulihkan snapshot yang diarsipkan secara permanen

Gunakan Restore-EC2SnapshotTier cmdlet.

```
Restore-EC2SnapshotTier `
    -SnapshotId snap-0abcdef1234567890 `
    -PermanentRestore $true
```

Untuk memulihkan snapshot yang diarsipkan sementara

Gunakan Restore-EC2SnapshotTier cmdlet.

```
Restore-EC2SnapshotTier `
    -SnapshotId snap-0abcdef1234567890 `
    -TemporaryRestoreDays 5
```

Ubah periode pemulihan untuk snapshot Amazon EBS yang dipulihkan sementara

Saat Anda memulihkan snapshot sementara, Anda harus menentukan jumlah hari di mana snapshot akan tetap dipulihkan di akun Anda. Setelah periode kedaluwasa, snapshot secara otomatis dihapus dari tingkat standar.

Anda dapat mengubah periode pemulihan untuk snapshot yang dipulihkan sementara kapan saja.

Anda dapat memilih untuk menambah atau mengurangi periode pemulihan, atau Anda dapat mengubah jenis pemulihan dari sementara menjadi permanen.

Jika Anda mengubah periode pemulihan, periode pemulihan baru berlaku sejak tanggal saat ini. Misalnya, jika Anda menentukan periode pemulihan baru 5 hari, snapshot akan tetap dipulihkan selama lima hari dari tanggal saat ini.

> **Note**
>
> Anda dapat mengakhiri pemulihan sementara lebih awal dengan mengatur periode pemulihan menjadi 1 hari.

Jika Anda mengubah jenis pemulihan dari sementara ke permanen, salinan snapshot dihapus dari tingkat arsip, dan snapshot tetap tersedia di akun Anda sampai Anda mengarsipkan ulang atau menghapusnya secara manual.

Console

Untuk memodifikasi periode pemulihan atau jenis pemulihan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Dalam daftar snapshot, pilih snapshot yang sebelumnya Anda pulihkan sementara, kemudian pilih Tindakan, Pulihkan snapshot dari arsip.
4. Untuk Jenis pemulihan, lakukan salah satu langkah berikut:
   - Untuk mengubah jenis pemulihan dari sementara ke permanen, pilih Permanen.
   - Untuk menambah atau mengurangi periode pemulihan, tetap pilih Sementara, dan kemudian untuk Periode pemulihan sementara, masukkan periode pemulihan baru dalam beberapa hari.
5. Untuk mengonfirmasi, pilih Pulihkan snapshot.

AWS CLI

Untuk memodifikasi periode pemulihan atau jenis pemulihan

Gunakan perintah restore- snapshot-tier. Untuk `--snapshot-id`, tentukan ID snapshot yang sebelumnya Anda pulihkan sementara. Untuk mengubah jenis pemulihan dari sementara ke permanen, tentukan `--permanent-restore` dan hilangkan `--temporary-restore-days`.

Untuk menambah atau mengurangi periode pemulihan, hilangkan `--permanent-restore` dan untuk `--temporary-restore-days`, tentukan periode pemulihan baru dalam hitungan hari.

Contoh: Menambah atau mengurangi periode pemulihan

Perintah berikut mengubah periode pemulihan untuk snapshot yang ditentukan menjadi beberapa `10` hari.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-0abcdef1234567890 \
    --temporary-restore-days 10
```

Berikut ini adalah output contoh.

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "RestoreDuration": 10,
    "IsPermanentRestore": false
}
```

Contoh: Ubah jenis pemulihan menjadi permanen

Perintah berikut mengubah jenis pemulihan untuk snapshot yang ditentukan dari sementara menjadi permanen.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-0abcdef1234567890 \
    --permanent-restore
```

Berikut ini adalah output contoh.

```
{
    "SnapshotId": "snap-0abcdef1234567890",
    "IsPermanentRestore": true
}
```

PowerShell

Untuk memodifikasi periode pemulihan atau jenis pemulihan

Gunakan Restore-EC2SnapshotTier cmdlet. Untuk `-SnapshotId`, tentukan ID snapshot yang sebelumnya Anda pulihkan sementara. Untuk mengubah jenis pemulihan dari sementara ke

permanen, tentukan `-PermanentRestore` dan hilangkan `-TemporaryRestoreDays`. Untuk menambah atau mengurangi periode pemulihan, hilangkan `-PermanentRestore` dan untuk `-TemporaryRestoreDays`, tentukan periode pemulihan baru dalam hitungan hari.

Contoh: Menambah atau mengurangi periode pemulihan

Perintah berikut mengubah periode pemulihan untuk snapshot yang ditentukan menjadi beberapa `10` hari.

```
Restore-EC2SnapshotTier `
    -SnapshotId snap-0abcdef1234567890 `
    -TemporaryRestoreDays 10
```

Contoh: Ubah jenis pemulihan menjadi permanen

Perintah berikut mengubah jenis pemulihan untuk snapshot yang ditentukan dari sementara menjadi permanen.

```
Restore-EC2SnapshotTier `
    -SnapshotId snap-0abcdef1234567890 `
    -PermanentRestore $true
```

Lihat snapshot Amazon EBS yang diarsipkan

Console

Untuk melihat informasi tingkat penyimpanan untuk snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Dalam daftar snapshot, pilih snapshot dan pilih tab Tingkat penyimpanan.

   Memberikan informasi berikut ini:

   - Perubahan tingkat terakhir dimulai pada — Tanggal dan waktu ketika arsip atau pemulihan terakhir dimulai.
   - Kemajuan perubahan tingkat - Kemajuan arsip terakhir atau tindakan pemulihan, sebagai persentase.

- Tingkat penyimpanan — Tingkat penyimpanan untuk snapshot. Selalu `archive` untuk snapshot yang diarsipkan, dan `standard` untuk snapshot yang disimpan di tingkat standar, termasuk snapshot yang dipulihkan sementara.
- Status tingkatan — Status arsip terakhir atau tindakan pemulihan.
- Arsip selesai pada — Tanggal dan waktu ketika arsip selesai.
- Pemulihan sementara kedaluwarsa pada — Tanggal dan waktu ketika snapshot yang dipulihkan sementara akan kedaluwarsa.

AWS CLI

Untuk melihat informasi pengarsipan tentang snapshot yang diarsipkan

Gunakan perintah describe-snap shot-tier-status. Tentukan filter `snapshot-id`, dan untuk nilai filter, tentukan ID snapshot. Atau, untuk melihat semua snapshot yang diarsipkan, hilangkan filter.

```
--filters "Name=snapshot-id, Values=snapshot_id"
```

Output mencakup parameter respons berikut:

- `Status` — Status snapshot. Selalu `completed` untuk snapshot yang diarsipkan. Hanya snapshot yang ada dalam status `completed` yang dapat diarsipkan.
- `LastTieringStartTime`— Tanggal dan waktu proses arsip dimulai, dalam format waktu UTC (YYYY-MM-DDTHH:MM:SSZ).
- `LastTieringOperationState` — Status proses arsip saat ini. Status yang mungkin termasuk: `archival-in-progress` `archival-completed` `archival-failed` | `permanent-restore-in-progress` | `permanent-restore-completed` | | `permanent-restore-failed` | `temporary-restore-in-progress` | `temporary-restore-completed` | `temporary-restore-failed`
- `LastTieringProgress` — Kemajuan proses arsip snapshot, dalam persentase.
- `StorageTier` — Tingkat penyimpanan untuk snapshot. Selalu `archive` untuk snapshot yang diarsipkan, dan `standard` untuk snapshot yang disimpan di tingkat standar, termasuk snapshot yang dipulihkan sementara.
- `ArchivalCompleteTime`— Tanggal dan waktu penyelesaian proses arsip, dalam format waktu UTC (YYYY-MM-DDTHH:MM:SSZ).

Contoh: Jelaskan snapshot

Contoh berikut menampilkan informasi tentang snapshot yang ditentukan.

```
aws ec2 describe-snapshot-tier-status \
    --filters "Name=snapshot-id, Values=snap-0abcdef1234567890"
```

Berikut ini adalah output contoh.

```
{
    "SnapshotTierStatuses": [
        {
            "Status": "completed",
            "ArchivalCompleteTime": "2021-09-15T17:33:16.147Z",
            "LastTieringProgress": 100,
            "Tags": [],
            "VolumeId": "vol-01234567890abcdef",
            "LastTieringOperationState": "archival-completed",
            "StorageTier": "archive",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-0abcdef1234567890",
            "LastTieringStartTime": "2021-09-15T16:44:37.574Z"
        }
    ]
}
```

Untuk melihat snapshot yang diarsipkan dan tingkat standar

Gunakan perintah describe-snapshots. Untuk`--snapshot-ids`, tentukan ID snapshot.

```
aws ec2 describe-snapshots --snapshot-ids snap-0abcdef1234567890
```

Berikut ini adalah output contoh. Parameter respons `StorageTier` menunjukkan apakah snapshot saat ini diarsipkan. `archive` menunjukkan bahwa snapshot saat ini diarsipkan dan disimpan di tingkat arsip, serta `standard` menunjukkan bahwa snapshot saat ini tidak diarsipkan dan disimpan di tingkat standar.

Dalam contoh output berikut, hanya `Snap A` yang diarsipkan. `Snap B` dan `Snap C` tidak diarsipkan.

Selain itu, parameter respons `RestoreExpiryTime` dikembalikan hanya untuk snapshot yang dipulihkan sementara dari arsip. Hal ini menunjukkan waktu snapshot yang dipulihkan sementara

akan dihapus secara otomatis dari tingkat standar. Parameter respons tersebut tidak dikembalikan untuk snapshot yang dipulihkan secara permanen.

Dalam contoh berikut output, `Snap C` dipulihkan sementara, dan akan secara otomatis dihapus dari tingkat standar pada 2021-09-19 T21:00:00.000Z (19 September 2021 pukul 21:00 UTC).


```
{
    "Snapshots": [
        {
            "Description": "Snap A",
            "Encrypted": false,
            "VolumeId": "vol-01234567890aaaaaa",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2021-09-07T21:00:00.000Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-01234567890aaaaaa",
            "StorageTier": "archive",
            "Tags": []
        },
        {
            "Description": "Snap B",
            "Encrypted": false,
            "VolumeId": "vol-09876543210bbbbbb",
            "State": "completed",
            "VolumeSize": 10,
            "StartTime": "2021-09-14T21:00:00.000Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-09876543210bbbbbb",
            "StorageTier": "standard",
            "RestoreExpiryTime": "2019-09-19T21:00:00.000Z",
            "Tags": []
        },
        {
            "Description": "Snap C",
            "Encrypted": false,
            "VolumeId": "vol-054321543210cccccc",
            "State": "completed",
            "VolumeSize": 12,
            "StartTime": "2021-08-01T21:00:00.000Z",
            "Progress": "100%",
```

```
            "OwnerId": "123456789012",
            "SnapshotId": "snap-054321543210cccccc",
            "StorageTier": "standard",
            "Tags": []
        }
    ]
}
```

Untuk hanya melihat snapshot yang disimpan di tingkat arsip atau tingkat standar

Gunakan perintah describe-snapshots. Sertakan `--filter` opsi, dengan nama filter`storage-tier`. Untuk nilai filter tentukan salah satu `archive` atau`standard`. Contoh berikut hanya menampilkan snapshot yang diarsipkan.

```
aws ec2 describe-snapshots --filters "Name=storage-tier,Values=archive"
```

PowerShell

Untuk melihat informasi pengarsipan tentang snapshot yang diarsipkan

Gunakan Get-EC2SnapshotTierStatus cmdlet.

```
Get-EC2SnapshotTierStatus `
    -Filter @{Name="snapshot-id"; Values="snap-0abcdef1234567890"}
```

Untuk melihat snapshot yang diarsipkan dan tingkat standar

Gunakan Get-EC2Snapshot cmdlet.

```
Get-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890
```

Untuk hanya melihat snapshot yang disimpan di tingkat arsip atau tingkat standar

Gunakan Get-EC2Snapshot cmdlet. Contoh berikut hanya menampilkan snapshot yang diarsipkan.

```
Get-EC2Snapshot `
    -Filter @{Name="storage-tier"; Values="archive"}
```

Memantau pengarsipan snapshot Amazon EBS menggunakan Acara CloudWatch

Amazon EBS memancarkan peristiwa yang terkait dengan tindakan pengarsipan snapshot. Anda dapat menggunakan AWS Lambda dan Amazon Ev CloudWatch ents untuk menangani pemberitahuan acara secara terprogram. Peristiwa dipancarkan atas dasar upaya terbaik. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Peristiwa berikut ini tersedia:

- `archiveSnapshot` — Dipancarkan ketika tindakan pengarsipan snapshot berhasil atau gagal.

  Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan arsip snapshot berhasil.

```
{
   "version": "0",
   "id": "01234567-0123-0123-0123-012345678901",
   "detail-type": "EBS Snapshot Notification",
   "source": "aws.ec2",
   "account": "123456789012",
   "time": "2021-05-25T13:12:22Z",
   "region": "us-east-1",
   "resources": [
     "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
   ],
   "detail": {
     "event": "archiveSnapshot",
     "result": "succeeded",
     "cause": "",
     "request-id": "123456789",
     "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
     "startTime": "2021-05-25T13:12:22Z",
     "endTime": "2021-05-45T15:30:00Z",
     "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
```

  Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan arsip snapshot gagal.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
```

```
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "archiveSnapshot",
    "result": "failed",
    "cause": "Source snapshot ID is not valid",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-05-45T15:30:00Z",
    "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
}
```

- `permanentRestoreSnapshot` — Dipancarkan ketika tindakan pemulihan permanen berhasil atau gagal.

  Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan pemulihan permanen berhasil.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "permanentRestoreSnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-10-45T15:30:00Z"
  }
```

```
}
```

Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan pemulihan permanen gagal.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "permanentRestoreSnapshot",
    "result": "failed",
    "cause": "Source snapshot ID is not valid",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-05-45T15:30:00Z",
    "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
}
```

- `temporaryRestoreSnapshot` — Dipancarkan ketika tindakan pemulihan sementara berhasil atau gagal.

  Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan pemulihan sementara berhasil.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
```

```
  "detail": {
    "event": "temporaryRestoreSnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-05-45T15:30:00Z",
    "restoreExpiryTime": "2021-06-45T15:30:00Z",
    "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
}
```

Berikut ini adalah contoh peristiwa yang dipancarkan saat tindakan pemulihan sementara gagal.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "temporaryRestoreSnapshot",
    "result": "failed",
    "cause": "Source snapshot ID is not valid",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-05-45T15:30:00Z",
    "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
}
```

- `restoreExpiry` — Dipancarkan saat periode pemulihan untuk snapshot yang dipulihkan sementara kedaluwarsa.

  Berikut adalah contohnya.

```
{
   "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2021-05-25T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "restoryExpiry",
    "result": "succeeded",
    "cause": "",
    "request-id": "1234567890",
    "snapshot_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
    "startTime": "2021-05-25T13:12:22Z",
    "endTime": "2021-05-45T15:30:00Z",
    "recycleBinExitTime": "2021-10-45T15:30:00Z"
  }
}
```

Hapus snapshot Amazon EBS

Setelah Anda tidak lagi memerlukan snapshot Amazon EBS untuk suatu volume, Anda dapat menghapusnya. Menghapus snapshot tidak berpengaruh pada volume. Menghapus volume tidak berdampak pada snapshot yang dibuat.

Topik

- Pertimbangan untuk menghapus snapshot
- Cara kerja menghapus snapshot inkremental
- Menghapus snapshot
- Hapus snapshot multi-volume

Pertimbangan untuk menghapus snapshot

Pertimbangan berikut berlaku untuk menghapus snapshot:

- Anda tidak dapat menghapus snapshot dari perangkat root volume EBS yang digunakan oleh AMI terdaftar. Pertimbangan ini berlaku bahkan jika AMI yang terdaftar diusangkan atau dinonaktifkan. Anda harus membatalkan pendaftaran AMI terlebih dahulu sebelum Anda dapat menghapus snapshot. Untuk informasi selengkapnya, lihat Mengh apus Pendaftaran AMI Anda.
- Anda tidak dapat menghapus snapshot yang dikelola oleh AWS Backup layanan menggunakan Amazon EC2. Sebagai gantinya, gunakan AWS Backup untuk menghapus titik pemulihan yang sesuai di brankas cadangan. Untuk informasi selengkapnya, lihat Menghapus cadangan dalam Panduan Developer AWS Backup .
- Anda dapat membuat,mempertahankan, dan menghapus snapshot secara manual, atau Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengelola snapshot Anda. Untuk informasi selengkapnya, lihat Amazon Data Lifecycle Manager.
- Meskipun Anda dapat menghapus snapshot yang masih dalam proses, snapshot harus selesai sebelum penghapusan diterapkan. Ini mungkin memerlukan waktu lama. Jika Anda juga berada di batas snapshot yang sama, dan Anda mencoba mengambil snapshot tambahan, Anda akan menemui kesalahan `ConcurrentSnapshotLimitExceeded`. Untuk informasi selengkapnya, lihat Kuota Layanan untuk Amazon EBS di Referensi Umum Amazon Web Services.
- Jika Anda menghapus snapshot yang cocok dengan aturan retensi Recycle Bin, snapshot akan disimpan di Recycle Bin alih-alih segera dihapus. Untuk informasi selengkapnya, lihat Recycle Bin.
- Anda tidak dapat menghapus snapshot yang terkait dengan AMI yang dinonaktifkan EBS-backed . Untuk informasi selengkapnya, lihat Menonaktifkan AMI.
- Anda tidak dapat menghapus snapshot yang dibagikan dengan Anda.
- Jika Anda menghapus snapshot bersama yang Anda miliki, semua akun yang digunakan untuk berbagi snapshot kehilangan akses ke snapshot tersebut.

Cara kerja menghapus snapshot inkremental

Jika Anda membuat snapshot berkala untuk volume, snapshotnya bersifat inkremental. Hal ini berarti bahwa hanya blok pada perangkat yang berubah setelah snapshot terbaru Anda disimpan di snapshot baru. Meskipun snapshot disimpan secara bertahap, proses penghapusan snapshot dirancang agar Anda hanya mempertahankan snapshot terbaru untuk membuat volume.

Jika data terdapat pada volume yang disimpan dalam snapshot atau serangkaian snapshot sebelumnya, dan data lalu dihapus dari volume tersebut di lain waktu, data tersebut masih dianggap sebagai data unik dari snapshot sebelumnya. Data unik hanya dihapus dari urutan snapshot jika semua snapshot yang mengacu pada data unik tersebut dihapus.

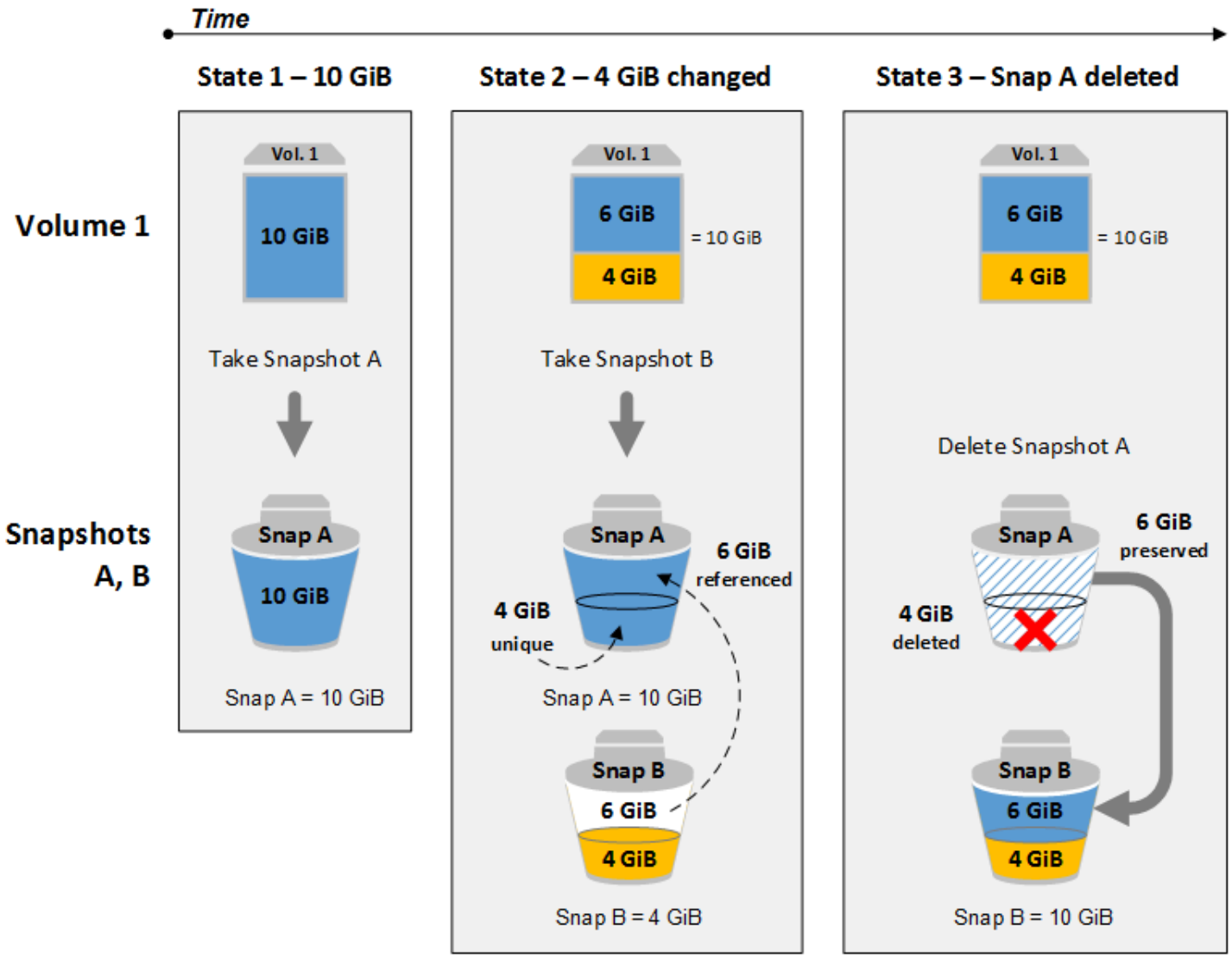
Saat Anda menghapus snapshot, hanya data yang dirujuk secara eksklusif oleh snapshot tersebut yang dihapus. Data unik hanya dihapus jika semua snapshot yang mereferensikannya dihapus. Menghapus snapshot sebelumnya dari suatu volume tidak akan memengaruhi kemampuan Anda untuk membuat volume dari snapshot yang lebih baru dari volume tersebut.

Menghapus snapshot mungkin tidak akan mengurangi biaya penyimpanan data organisasi Anda. Snapshot lain mungkin merujuk pada data snapshot, dan data yang direferensikan selalu dipertahankan. Jika Anda menghapus snapshot yang berisi data yang digunakan berikutnya, biaya yang terkait dengan data yang direferensikan dialokasikan ke snapshot berikutnya. Untuk informasi selengkapnya tentang cara snapshot menyimpan data, lihat Cara kerja snapshot Amazon EBS dan contoh berikut.

Dalam diagram berikut, Volume 1 ditampilkan pada tiga titik waktu. Snapshot telah menangkap masing-masing dari dua status pertama, dan di status ketiga, snapshot telah dihapus.

- Dalam keadaan 1, volume memiliki 10 GiB data. Karena Snap A adalah snapshot pertama yang diambil dari volume, secara keseluruhan 10 GiB data harus disalin. Dalam keadaan ini, Anda dikenakan biaya untuk menyimpan 10 GiB data snapshot.
- Dalam keadaan 2, volume masih berisi 10 GiB data, tetapi 4 GiB telah berubah. Snap B hanya menyimpan 4 GiB yang berubah setelah Snap A diambil, dan merujuk 6 GiB data tidak berubah yang sudah disimpan di Snap A. Dalam keadaan ini, Anda dikenakan biaya untuk menyimpan 14 GiB data snapshot (10 Gib dari Snap A + 4 GiB dari Snap B).
- Dalam keadaan 3, volume tidak berubah tetapi Snap A dihapus. Karena 6 GiB data yang tidak berubah di Snap A masih direferensikan oleh Snap B, data tersebut dipertahankan dan dikaitkan dengan Snap B. 4 GiB data unik di Snap A dihapus karena tidak lagi direferensikan oleh snapshot lain. Dalam keadaan ini, Anda dikenakan biaya untuk menyimpan 10 GiB data snapshot (6 GiB data yang disimpan dari Snap A + 4 GiB data di Snap B).

Menghapus snapshot dengan beberapa data yang direferensikan oleh snapshot lain



Menghapus snapshot

Console

Untuk menghapus snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot.
4. Pilih Tindakan, Hapus snapshot.
5. Saat diminta konfirmasi, masukkan **`delete`**, lalu pilih Hapus.

AWS CLI

Untuk menghapus snapshot

Gunakan perintah delete-snapshot.

```
aws ec2 delete-snapshot --snapshot-id snap-0abcdef1234567890
```

PowerShell

Untuk menghapus snapshot

Gunakan Remove-EC2Snapshot cmdlet.

```
Remove-EC2Snapshot -SnapshotId snap-0abcdef1234567890
```

Tip pemecahan masalah

Jika Anda mendapatkan `Failed to delete snapshot` kesalahan yang menunjukkan bahwa snapshot saat ini digunakan oleh AMI, Anda harus mem batalkan pendaftaran AMI terkait sebelum Anda dapat menghapus snapshot. Anda tidak dapat menghapus snapshot yang terkait dengan AMI.
Jika Anda menggunakan konsol dan AMI terkait dinonaktifkan, Anda harus memilih filter Gambar dinonaktifkan di layar AMI untuk melihat AMI yang dinonaktifkan.

Hapus snapshot multi-volume

Untuk menghapus snapshot multivolume, ambil semua snapshot untuk snapshot multivolume Anda menggunakan tanda yang Anda terapkan ke set saat Anda membuat snapshot. Lalu, hapus snapshot secara terpisah.

Anda tidak akan dicegah menghapus snapshot individual dalam set snapshot multi-volume. Jika Anda menghapus snapshot saat berada di `pending state`, hanya snapshot tersebut yang dihapus. Snapshot lain dalam set snapshot multivolume masih berhasil diselesaikan.

Pemulihan snapshot cepat Amazon EBS

Pemulihan snapshot cepat (FSR) Amazon EBS memungkinkan Anda membuat volume dari snapshot yang sepenuhnya diinisialisasi saat pembuatan. Ini menghilangkan latensi I/O operasi pada blok ketika diakses untuk pertama kalinya. Volume yang dibuat menggunakan pemulihan snapshot cepat secara instans memberikan semua performa yang disediakan.

Untuk memulai, aktifkan pemulihan snapshot cepat untuk snapshot tertentu di Zona Ketersediaan tertentu. Setiap pasangan snapshot dan Zona Ketersediaan merujuk pada satu pemulihan snapshot cepat. Saat Anda membuat volume dari salah satu snapshot ini di salah satu Zona Ketersediaan yang diaktifkan, volume tersebut dipulihkan menggunakan pemulihan snapshot cepat.

Anda harus secara eksplisit mengaktifkan pemulihan snapshot cepat untuk setiap snapshot. Misalnya, jika Anda membuat snapshot baru dari volume yang dipulihkan dari snapshot yang diaktifkan pemulihan snapshot cepat, snapshot baru tidak diaktifkan secara otomatis untuk pemulihan snapshot cepat. Jika Anda menyalin snapshot yang diaktifkan untuk pemulihan snapshot cepat, salinan snapshot tidak secara otomatis diaktifkan untuk pemulihan snapshot cepat.

Jumlah volume yang dapat Anda pulihkan dengan manfaat performa penuh dari pemulihan snapshot cepat ditentukan oleh kredit pembuatan volume untuk snapshot tersebut. Untuk informasi selengkapnya, lihat Kredit pembuatan volume pemulihan snapshot cepat Amazon EBS.

Anda dapat mengaktifkan pemulihan snapshot cepat untuk snapshot yang Anda miliki dan untuk snapshot publik serta privat yang dibagikan dengan Anda.

Daftar Isi

- Pertimbangan-pertimbangan
- Harga dan Penagihan
- Kredit pembuatan volume pemulihan snapshot cepat Amazon EBS
- Konfigurasikan pemulihan snapshot cepat untuk snapshot Amazon EBS
- Periksa status pemulihan snapshot cepat untuk snapshot Amazon EBS
- Lihat volume Amazon EBS yang dipulihkan menggunakan pemulihan snapshot cepat

Pertimbangan-pertimbangan

- Pemulihan snapshot cepat tidak didukung dengan AWS Outposts, Zona Lokal, dan Zona Panjang Gelombang.

- Pemulihan snapshot cepat dapat diaktifkan pada snapshot dengan ukuran 16 TiB atau kurang.
- Volume yang disediakan dengan kinerja hingga 64.000 IOPS dan 1.000 MiB/s throughput menerima manfaat kinerja penuh dari pemulihan snapshot cepat. Untuk volume yang disediakan dengan kinerja lebih dari 64.000 IOPS atau 1.000 MiB/s throughput, sebaiknya inisialisasi volume untuk menerima kinerja penuhnya.
- Anda dapat mengaktifkan hingga 5 snapshot untuk pemulihan snapshot cepat per Wilayah. Kuota berlaku untuk snapshot yang Anda miliki dan snapshot yang dibagikan dengan Anda. Jika Anda mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibagikan kepada Anda, itu dihitung terhadap kuota pemulihan snapshot cepat Anda. Ini tidak termasuk dalam kuota pemulihan snapshot cepat pemilik snapshot.
- Amazon EBS memancarkan CloudWatch peristiwa Amazon saat status pemulihan snapshot cepat untuk snapshot berubah. Untuk informasi selengkapnya, lihat Peristiwa pemulihan snapshot cepat EBS.

Harga dan Penagihan

Anda dikenai biaya untuk setiap menit pengaktifan pemulihan snapshot cepat untuk snapshot dalam Zona Ketersediaan tertentu. Biaya bersifat pro-rata minimal satu jam.

Misalnya, jika Anda mengaktifkan pemulihan snapshot cepat untuk satu snapshot di `US-East-1a` selama satu bulan (30 hari), Anda dikenai biaya 540 USD (`1` snapshot x `1` AZ x `720` jam x `$0.75` per jam). Jika Anda mengaktifkan pemulihan snapshot cepat untuk dua snapshot dalam `us-east-1a`, `us-east-1b`, dan `us-east-1c` untuk periode yang sama, Anda dikenai biaya 3240 USD (`2` snapshot x `3` AZ x `720` jam x `$0.75` per jam).

Jika Anda mengaktifkan pemulihan snapshot cepat untuk snapshot publik atau privat yang dibagikan dengan Anda, akun Anda dikenai biaya; pemilik snapshot tidak dikenai biaya. Ketika snapshot yang dibagikan dengan Anda dihapus atau tidak dibagikan oleh pemilik snapshot, pemulihan snapshot cepat dinonaktifkan untuk snapshot di akun Anda dan penagihan dihentikan.

Untuk informasi selengkapnya, lihat harga Amazon EBS.

Kredit pembuatan volume pemulihan snapshot cepat Amazon EBS

Jumlah volume yang menerima manfaat kinerja penuh dari pemulihan snapshot cepat ditentukan oleh kredit pembuatan volume untuk snapshot tersebut. Ada satu bucket kredit per snapshot per Zona Ketersediaan. Setiap volume yang Anda buat dari snapshot dengan pemulihan snapshot cepat yang diaktifkan akan menggunakan satu kredit dari bucket kredit. Anda harus memiliki setidaknya

satu kredit dalam bucket untuk membuat volume yang diinisialisasi dari snapshot. Jika Anda membuat volume tetapi ada kurang dari satu kredit dalam bucket, volume dibuat tanpa manfaat pemulihan snapshot cepat.

Saat Anda mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibagikan kepada Anda, Anda mendapatkan bucket kredit terpisah untuk snapshot yang dibagikan dalam akun Anda. Jika Anda membuat volume dari snapshot yang dibagikan, kredit digunakan dari bucket Anda; kredit tidak digunakan dari bucket kredit pemilik snapshot.

Ukuran bucket kredit dan tingkat isi ulang didasarkan pada ukuran snapshot (yang juga merupakan ukuran volume sumber), bukan ukuran data snapshot. Misalnya, jika Anda membuat snapshot dari volume 200 GiB yang memiliki 150 GiB data, dan mengaktifkannya untuk pemulihan snapshot cepat, ukuran bucket kredit dan tingkat isi ulang didasarkan pada 200 GiB.

Saat Anda mengaktifkan pemulihan snapshot cepat untuk snapshot, bucket kredit dimulai dengan nol kredit, dan akan diisi pada tingkat yang ditetapkan hingga mencapai kapasitas kredit maksimumnya. Selain itu, saat Anda menggunakan kredit, bucket kredit diisi ulang dari waktu ke waktu hingga mencapai kapasitas kredit maksimumnya.

Laju pengisian untuk bucket dihitung sebagai berikut:

```
MIN (10, (1024 ÷ snapshot_size_gib))
```

Dan ukuran bucket kredit dihitung sebagai berikut:

```
MAX (1, MIN (10, (1024 ÷ snapshot_size_gib)))
```

Misalnya, jika Anda mengaktifkan pemulihan snapshot cepat untuk snapshot dengan ukuran 128 GiB, tingkat pengisian adalah 0.1333 kredit per menit.

```
MIN (10, (1024 ÷ 128))
 = MIN (10, 8)
 = 8 credits per hour
 = 0.1333 credits per minute
```

Dan ukuran maksimum bucket kredit adalah 8 kredit.

```
MAX (1, MIN (10, (1024 ÷ 128)))
 = MAX (1, MIN (10, 8))
 = MAX (1, 8)
```

```
= 8 credits
```

Dalam contoh ini, saat Anda mengaktifkan pemulihan snapshot cepat, bucket kredit dimulai dengan nol kredit. Setelah 8 menit, bucket kredit memiliki kredit yang cukup untuk membuat satu volume (`0.1333 credits × 8 minutes = 1.066 credits`) yang diinisialisasi. Saat bucket kredit penuh, Anda dapat membuat 8 volume yang diinisialisasi secara bersamaan (8 kredit). Ketika bucket berada di bawah kapasitas maksimumnya, bucket diisi ulang dengan `0.1333` kredit per menit.

Anda dapat menggunakan CloudWatch metrik untuk memantau ukuran bucket kredit Anda dan jumlah kredit yang tersedia di setiap bucket. Untuk informasi selengkapnya, lihat Metrik untuk pemulihan snapshot cepat.

Setelah Anda membuat volume dari pemulihan snapshot dengan pemulihan snapshot cepat, Anda dapat menjelaskan volume menggunakan describe-volumes dan memeriksa bidang `fastRestored` di output untuk menentukan apakah volume dibuat sebagai volume menggunakan pemulihan snapshot cepat.

Konfigurasikan pemulihan snapshot cepat untuk snapshot Amazon EBS

Pemulihan snapshot cepat dinonaktifkan untuk snapshot secara default. Anda dapat mengaktifkan atau menonaktifkan pemulihan snapshot cepat untuk snapshot yang Anda miliki dan untuk snapshot yang dibagikan dengan Anda. Saat Anda mengaktifkan atau menonaktifkan pemulihan snapshot cepat untuk snapshot, perubahan hanya berlaku pada akun Anda.

> **Note**
>
> Saat Anda mengaktifkan pemulihan snapshot cepat untuk suatu snapshot, akun Anda akan dikenai biaya untuk setiap menit pengaktifan pemulihan snapshot cepat di Zona Ketersediaan tertentu. Biaya bersifat pro-rata dan memiliki minimal satu jam.

Saat Anda menghapus snapshot yang Anda miliki, pemulihan snapshot cepat secara otomatis dinonaktifkan untuk snapshot tersebut di akun Anda. Jika Anda mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibagikan kepada Anda, dan pemilik snapshot menghapus atau membatalkan pembagiannya, pemulihan snapshot cepat secara otomatis dinonaktifkan untuk snapshot yang dibagikan di akun Anda.

Jika Anda mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibagikan kepada Anda, dan snapshot tersebut telah dienkripsi menggunakan CMK kustom, pemulihan snapshot cepat

tidak secara otomatis dinonaktifkan untuk snapshot saat pemilik snapshot mencabut akses ke CMK kustom. Anda harus menonaktifkan pemulihan snapshot cepat untuk snapshot ini secara manual.

Setelah Anda mengaktifkan pemulihan snapshot cepat untuk snapshot, snapshot memasuki status `optimizing`. Snapshot yang ada dalam status `optimizing` memberikan beberapa manfaat performa saat menggunakannya untuk memulihkan volume. Snapshot mulai memberikan manfaat performa penuh dari pemulihan snapshot cepat hanya setelah memasuki status `enabled`.

Console

Untuk mengkonfigurasi pemulihan snapshot cepat

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot, dan pilih Tindakan, Kelola pemulihan snapshot cepat.
4. Bagian Pengaturan pemulihan snapshot cepat mencantumkan semua Zona Ketersediaan di mana Anda dapat mengaktifkan pemulihan snapshot cepat untuk snapshot yang dipilih. Volume status saat ini menunjukkan apakah pemulihan snapshot cepat saat ini diaktifkan atau dinonaktifkan untuk setiap zona.

   Untuk mengaktifkan pemulihan snapshot cepat di zona yang saat ini dinonaktifkan, pilih zona, pilih Aktifkan, lalu konfirmasi, pilih Aktifkan.

   Untuk menonaktifkan pemulihan snapshot cepat di zona yang saat ini diaktifkan, pilih zona, lalu pilih Nonaktifkan.
5. Setelah Anda membuat perubahan yang diperlukan, pilih Tutup.

AWS CLI

Untuk mengaktifkan pemulihan snapshot cepat

Gunakan perintah enable- fast-snapshot-restore.

```
aws ec2 enable-fast-snapshot-restores \
    --availability-zones us-east-1a us-east-1b \
    --source-snapshot-ids snap-0abcdef1234567890
```

Untuk menonaktifkan pemulihan snapshot cepat

Gunakan perintah disable- fast-snapshot-restore.

```
aws ec2 disable-fast-snapshot-restores \
    --availability-zones us-east-1a \
    --source-snapshot-ids snap-0abcdef1234567890
```

Contoh berikut menggunakan perintah describe-fast-snapshot-untuk menggambarkan pemulihan snapshot cepat yang dinonaktifkan.

```
aws ec2 describe-fast-snapshot-restores \
    --filters Name=state,Values=disabled
```

PowerShell

Untuk mengaktifkan pemulihan snapshot cepat

Gunakan Enable-EC2FastSnapshotRestore cmdlet.

```
Enable-EC2FastSnapshotRestore `
    -AvailabilityZone us-east-1a us-east-1b `
    -SourceSnapshotId snap-0abcdef1234567890
```

Untuk menonaktifkan pemulihan snapshot cepat

Gunakan Disable-EC2FastSnapshotRestore cmdlet.

```
Disable-EC2FastSnapshotRestore `
    -AvailabilityZone us-east-1a `
    -SourceSnapshotId snap-0abcdef1234567890
```

Contoh berikut menggunakan Get-EC2FastSnapshotRestore cmdlet untuk menggambarkan pemulihan snapshot cepat yang dinonaktifkan.

```
Get-EC2FastSnapshotRestore `
    -Filter @{Name="state"; Values="disabled"}
```

Periksa status pemulihan snapshot cepat untuk snapshot Amazon EBS

Pemulihan snapshot cepat untuk snapshot dapat berada di salah satu status berikut.

- `enabling` – Permintaan dibuat untuk mengaktifkan pemulihan snapshot cepat.
- `optimizing` — Pemulihan snapshot cepat sedang diaktifkan. Akan memakan waktu 60 menit per TiB untuk mengoptimalkan snapshot. Snapshot dalam keadaan ini menawarkan beberapa manfaat performa saat memulihkan volume.
- `enabled` — Pemulihan snapshot cepat sedang diaktifkan. Snapshot yang berada dalam keadaan ini dan memiliki kredit pembuatan volume yang memadai menawarkan manfaat performa penuh saat memulihkan volume.
- `disabling` — Permintaan dibuat untuk menonaktifkan pemulihan snapshot cepat, atau permintaan untuk mengaktifkan pemulihan snapshot cepat yang gagal.
- `disabled` — Pemulihan snapshot cepat sedang dinonaktifkan. Anda dapat mengaktifkan pemulihan snapshot cepat sesuai kebutuhan.

Anda dapat melihat status pemulihan snapshot cepat untuk snapshot yang Anda miliki atau untuk snapshot yang dibagikan dengan Anda.

Console

Untuk melihat keadaan pemulihan snapshot cepat

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot.
4. Di tab Detail, Pemulihan snapshot cepat, menunjukkan status pemulihan snapshot cepat.

AWS CLI

Untuk melihat snapshot dengan pemulihan snapshot cepat diaktifkan

Gunakan perintah describe-fast- snapshot-.

```
aws ec2 describe-fast-snapshot-restores --filters Name=state,Values=enabled
```

Berikut ini adalah output contoh.

```
{
    "FastSnapshotRestores": [
        {
```

```
            "SnapshotId": "snap-0e946653493cb0447",
            "AvailabilityZone": "us-east-2a",
            "State": "enabled",
            "StateTransitionReason": "Client.UserInitiated - Lifecycle state
 transition",
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.596Z",
            "OptimizingTime": "2020-01-25T23:58:25.573Z",
            "EnabledTime": "2020-01-25T23:59:29.852Z"
        },
        {
            "SnapshotId": "snap-0e946653493cb0447",
            "AvailabilityZone": "us-east-2b",
            "State": "enabled",
            "StateTransitionReason": "Client.UserInitiated - Lifecycle state
 transition",
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.596Z",
            "OptimizingTime": "2020-01-25T23:58:25.573Z",
            "EnabledTime": "2020-01-25T23:59:29.852Z"
        }
    ]
}
```

PowerShell

Untuk melihat snapshot dengan pemulihan snapshot cepat diaktifkan

Gunakan Get-EC2FastSnapshotRestore cmdlet.

```
Get-EC2FastSnapshotRestore `
    -Filter @{Name="state"; Values="enabled"}
```

Lihat volume Amazon EBS yang dipulihkan menggunakan pemulihan snapshot cepat

Saat Anda membuat volume dari snapshot yang diaktifkan untuk pemulihan snapshot cepat di Zona Ketersediaan untuk volume, snapshot akan dipulihkan menggunakan pemulihan snapshot cepat.

AWS CLI

Untuk melihat volume yang dibuat dari snapshot yang diaktifkan untuk pemulihan snapshot cepat

Gunakan perintah describe-volumes.

```
aws ec2 describe-volumes --filters Name=fast-restored,Values=true
```

Berikut ini adalah output contoh.

```
{
    "Volumes": [
        {
            "Attachments": [],
            "AvailabilityZone": "us-east-2a",
            "CreateTime": "2020-01-26T00:34:11.093Z",
            "Encrypted": true,
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/8c5b2c63-b9bc-45a3-
a87a-5513e232e843",
            "Size": 20,
            "SnapshotId": "snap-0abcdef1234567890",
            "State": "available",
            "VolumeId": "vol-01234567890abcdef",
            "Iops": 100,
            "VolumeType": "gp2",
            "FastRestored": true
        }
    ]
}
```

PowerShell

Untuk melihat volume yang dibuat dari snapshot yang diaktifkan untuk pemulihan snapshot cepat

Gunakan Get-EC2Volume cmdlet.

```
Get-EC2Volume -Filter @{Name="fast-restored"; Values="true"}
```

Kunci snapshot Amazon EBS

Anda dapat mengunci snapshot Amazon EBS untuk melindunginya dari penghapusan yang tidak disengaja atau berbahaya, atau menyimpannya dalam format WORM (write-once-read-many) untuk durasi tertentu. Sementara snapshot dikunci, snapshot tersebut tidak dapat dihapus oleh pengguna

mana pun, terlepas dari izin IAM mereka. Anda dapat terus menggunakan snapshot terkunci dengan cara yang sama ketika Anda akan menggunakan snapshot lainnya.

> **Note**
>
> Kunci snapshot telah dinilai oleh Cohasset Associates untuk digunakan di lingkungan yang tunduk pada peraturan SEC 17a-4, CFTC, dan FINRA. Untuk informasi selengkapnya tentang cara kunci snapshot terkait dengan peraturan ini, lihat Penilaian Kepatuhan Cohasset Associates.

Anda dapat mengunci snapshot dalam salah satu dari dua mode: mode kepatuhan atau mode tata kelola, dan snapshot dapat dikunci selama durasi tertentu atau hingga tanggal tertentu. Untuk informasi selengkapnya, lihat Mode Kunci dan Durasi kunci.

Harga

Anda dapat mengunci dan membuka snapshot tanpa biaya tambahan. Anda membayar biaya penyimpanan snapshot Amazon EBS standar untuk snapshot terkunci.

Topik

- Konsep kunci snapshot Amazon EBS
- Pertimbangan untuk kunci snapshot Amazon EBS
- Kontrol akses ke kunci snapshot Amazon EBS
- Mengunci snapshot Amazon EBS
- Membuka cuplikan snapshot Amazon EBS
- Perbarui pengaturan kunci snapshot Amazon EBS
- Pantau kunci snapshot Amazon EBS

Konsep kunci snapshot Amazon EBS

Berikut ini adalah konsep penting untuk dipahami saat Anda mulai menggunakan kunci snapshot.

Daftar Isi

- Mode Kunci

- Durasi kunci
- Cooling-off periode
- Status kunci

Mode Kunci

Anda dapat mengunci snapshot dalam salah satu dari dua mode:

Mode tata kelola

Setelah snapshot dikunci, pengguna dengan izin IAM yang sesuai dapat membuka kunci snapshot dan memodifikasi mode kunci dan durasi kunci atau tanggal kedaluwarsa kapan saja. Saat Anda mengunci snapshot dalam mode tata kelola, snapshot segera dikunci; tidak ada periode pendinginan. Untuk menghapus snapshot setelah dikunci dalam mode tata kelola, Anda harus membuka kunci snapshot terlebih dahulu atau Anda harus menunggu kunci kedaluwarsa.

Anda dapat menggunakan mode tata kelola untuk memenuhi persyaratan tata kelola data organisasi Anda dengan memastikan bahwa hanya pengguna tertentu yang memiliki izin untuk membuka kunci snapshot dan memodifikasi konfigurasi kunci snapshot. Anda juga dapat menggunakan mode tata kelola untuk menguji konfigurasi kunci Anda sebelum mengunci snapshot dalam mode kepatuhan.

Mode kepatuhan

Saat Anda mengunci snapshot dalam mode kepatuhan, Anda dapat secara opsional menentukan periode pendinginan yang dimulai segera setelah Anda mengunci snapshot. Selama periode pendinginan, pengguna dengan izin yang sesuai dapat membuka kunci snapshot, mengubah mode kunci, menambah atau mengurangi periode pendinginan, dan menambah atau mengurangi durasi kunci atau tanggal kedaluwarsa. Setelah periode pendinginan berakhir, Anda tidak dapat membuka kunci snapshot, mengubah mode kunci, atau mengurangi durasi kunci atau tanggal kedaluwarsa; Anda hanya dapat menambah durasi kunci atau tanggal kedaluwarsa. Untuk menghapus snapshot setelah dikunci sesuai dan periode pendinginan telah kedaluwarsa, Anda harus menunggu hingga kunci kedaluwarsa.

> **Note**
>
> Anda dapat mengunci snapshot dalam mode kepatuhan tanpa periode pendinginan dengan menghilangkan periode pendinginan dalam permintaan. Setelah periode pendinginan berakhir, Anda tidak dapat membuka kunci snapshot, mengubah mode kunci, atau

mengurangi durasi kunci atau tanggal kedaluwarsa; Anda hanya dapat menambah durasi kunci atau tanggal kedaluwarsa.

Anda dapat menggunakan mode kepatuhan untuk melindungi snapshot yang tidak boleh dihapus untuk periode tertentu karena alasan kepatuhan. Mode kepatuhan menawarkan manfaat sebagai berikut:

- Hal ini memungkinkan konfigurasi WORM (tulis sekali, baca banyak) untuk snapshot Anda.
- Ini memberikan lapisan pertahanan tambahan yang melindungi snapshot dari penghapusan yang tidak disengaja atau berbahaya.
- Ini memberlakukan periode retensi, yang mencegah penghapusan dini oleh pengguna istimewa, untuk memenuhi kebijakan dan prosedur perlindungan data organisasi Anda.

Note

Satu-satunya cara untuk menghapus snapshot yang terkunci dalam mode kepatuhan sebelum kuncinya berakhir adalah dengan menutup AWS akun terkait.

Durasi kunci

Durasi kunci adalah periode waktu di mana snapshot tetap terkunci. Anda dapat menentukan durasi kunci sebagai salah satu dari berikut ini, tetapi tidak keduanya:

Jumlah hari

Durasi kunci ditentukan sebagai beberapa hari di mana snapshot tetap terkunci. Setelah jumlah hari yang ditentukan berlalu, snapshot secara otomatis dibuka kuncinya. Durasi dapat berkisar dari 1 hari hingga 36500 hari (100 tahun).

Tanggal kedaluwarsa kunci

Durasi kunci ditentukan oleh tanggal kedaluwarsa di masa mendatang. Snapshot tetap terkunci sampai tanggal kedaluwarsa kunci tercapai. Ketika tanggal kedaluwarsa kunci tercapai, snapshot secara otomatis dibuka kuncinya.

Cooling-off periode

Periode pendinginan adalah periode waktu opsional yang dapat Anda tentukan saat Anda mengunci snapshot dalam mode kepatuhan. Selama periode pendinginan, pengguna dengan izin yang sesuai dapat membuka kunci snapshot, mengubah mode kunci, menambah atau mengurangi periode pendinginan, dan menambah atau mengurangi durasi kunci atau tanggal kedaluwarsa. Setelah periode pendinginan berakhir, pengguna tidak dapat membuka kunci snapshot, mengubah mode kunci, mengembalikan periode pendinginan, atau mengurangi durasi penguncian, terlepas dari izin mereka.

Snapshot tidak dapat dihapus selama periode pendinginan.

Jika ditentukan, periode pendinginan dimulai segera setelah Anda mengunci snapshot. Jika dihilangkan, snapshot dikunci dalam mode kepatuhan segera tanpa periode pendinginan.

Periode pendinginan dapat berkisar dari 1 hingga 72 jam. Anda dapat mengunci snapshot dalam mode kepatuhan tanpa periode pendinginan dengan menghilangkan periode pendinginan dalam permintaan.

Status kunci

Kunci snapshot dapat berada di salah satu status berikut:

- `compliance-cooloff` — Snapshot telah dikunci dalam mode kepatuhan, tetapi masih dalam periode pendinginan. Snapshot tidak dapat dihapus, tetapi dapat dibuka kuncinya dan pengaturan kunci dapat dimodifikasi oleh pengguna dengan izin yang sesuai.
- `governance` — Snapshot dikunci dalam mode tata kelola. Snapshot tidak dapat dihapus, tetapi dapat dibuka kuncinya dan pengaturan kunci dapat dimodifikasi oleh pengguna dengan izin yang sesuai.
- `compliance` — Snapshot dikunci dalam mode kepatuhan tanpa periode pendinginan atau periode pendinginan telah kedaluwarsa. Snapshot tidak dapat dibuka atau dihapus. Durasi kunci hanya dapat ditingkatkan oleh pengguna dengan izin yang sesuai.
- `expired` — Snapshot dikunci dalam mode kepatuhan atau tata kelola, tetapi kunci telah kedaluwarsa. Snapshot tidak terkunci dan dapat dihapus.

Pertimbangan untuk kunci snapshot Amazon EBS

Ingatlah hal berikut saat mengunci snapshot Amazon EBS.

- Anda dapat mengunci snapshot hanya jika snapshot ada dalam status `pending` atau `completed`.
  - Jika Anda mengunci snapshot saat berada dalam status `pending`, dan Anda menguncinya untuk durasi tertentu, durasi kunci hanya dimulai ketika snapshot mencapai status `completed`. Snapshot tidak dapat dihapus saat berada dalam status `pending`.
  - Jika Anda mengunci snapshot saat berada dalam status `pending` dan pembuatan snapshot gagal karena alasan apa pun, kunci dibatalkan.
- Jika Anda memperpanjang durasi penguncian untuk snapshot yang terkunci dalam mode kepatuhan setelah periode pendinginan berakhir, Anda tidak dapat menentukan periode pendinginan lainnya. Jika Anda menentukan periode pendinginan, permintaan gagal.
- Anda dapat mengunci snapshot yang diarsipkan. Dan Anda dapat mengarsipkan snapshot yang terkunci.
- Anda dapat mengunci snapshot yang terkait dengan AMI.
- Anda dapat membatalkan pendaftaran AMI yang terkait dengan snapshot yang dikunci.
- Anda dapat menghapus kunci KMS yang digunakan untuk mengenkripsi snapshot yang terkunci.
- Kami menyarankan Anda untuk tidak mengunci snapshot yang dibuat oleh AWS Backup. AWS Backup sudah memastikan bahwa snapshot-nya tidak dihapus sebelum periode retensinya berakhir. Untuk menambahkan lapisan keamanan tambahan untuk snapshot yang dikelola oleh AWS Backup, sebaiknya gunakan AWS Backup Vault Lock. Untuk informasi selengkapnya, lihat Kunci Penyimpanan AWS Backup.
- Anda tidak dapat mengunci snapshot selama pembuatan atau selama pendaftaran AMI.
- Anda tidak dapat mengunci snapshot Amazon EBS lokal di AWS Outposts.
- Satu-satunya cara untuk menghapus snapshot yang terkunci dalam mode kepatuhan sebelum kuncinya berakhir adalah dengan menutup AWS akun terkait.

  Jika Anda menutup AWS akun saat Anda telah mengunci snapshot, akun Anda ditang AWS guhkan selama 90 hari dengan snapshot Anda tetap utuh. Jika Anda tidak membuka kembali akun Anda dalam 90 hari, AWS hapus snapshot Anda, meskipun terkunci.

Kontrol akses ke kunci snapshot Amazon EBS

Secara default, pengguna tidak memiliki izin untuk bekerja dengan kunci snapshot. Untuk mengizinkan pengguna bekerja dengan sumber daya ini, Anda harus membuat kebijakan IAM yang memberikan izin menggunakan sumber daya dan tindakan API tertentu. Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Topik

- Izin yang diperlukan
- Batasi akses dengan kunci syarat

Izin yang diperlukan

Untuk bekerja dengan kunci snapshot, pengguna memerlukan izin berikut.

- `ec2:LockSnapshot` — Untuk mengunci snapshot.
- `ec2:UnlockSnapshot` — Untuk membuka kunci snapshot.
- `ec2:DescribeLockedSnapshots` — Untuk melihat pengaturan kunci snapshot.

Berikut ini adalah contoh kebijakan IAM yang memberi pengguna izin untuk mengunci dan membuka kunci snapshot, dan untuk melihat pengaturan kunci snapshot. Ini termasuk izin `ec2:DescribeSnapshots` untuk pengguna konsol. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSnapshotLockOperations",
      "Effect": "Allow",
      "Action": [
        "ec2:LockSnapshot",
        "ec2:UnlockSnapshot",
        "ec2:DescribeLockedSnapshots",
        "ec2:DescribeSnapshots"
      ],
      "Resource": [
        "arn:aws:ec2:*::snapshot/*",
        "arn:aws:ec2:*:111122223333:volume/*"
      ]
    }
  ]
}
```

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Batasi akses dengan kunci syarat

Anda dapat menggunakan kunci syarat untuk membatasi cara pengguna diizinkan untuk mengunci snapshot.

Topik

- ec2: SnapshotLockDuration
- ec2: CoolOffPeriod

ec2: SnapshotLockDuration

Anda dapat menggunakan kunci syarat `ec2:SnapshotLockDuration` untuk membatasi pengguna pada durasi kunci tertentu saat mengunci snapshot.

Contoh kebijakan berikut membatasi pengguna untuk menentukan durasi kunci antara `10` dan `50` hari.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSnapshotLockWithDurationCondition",
      "Effect": "Allow",
      "Action": "ec2:LockSnapshot",
      "Resource": "arn:aws:ec2:*::snapshot/*",
      "Condition": {
        "NumericGreaterThan": {
          "ec2:SnapshotLockDuration": 10
        },
        "NumericLessThan": {
          "ec2:SnapshotLockDuration": 50
        }
      }
    }
  ]
}
```

ec2: CoolOffPeriod

Anda dapat menggunakan kunci syarat `ec2:CoolOffPeriod` untuk mencegah pengguna mengunci snapshot dalam mode kepatuhan tanpa periode pendinginan.

Contoh kebijakan berikut membatasi pengguna untuk menentukan periode pendinginan lebih dari 48 jam saat mengunci snapshot dalam mode kepatuhan.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSnapshotLockWithCondition",
```

```
      "Effect": "Allow",
      "Action": "ec2:LockSnapshot",
      "Resource": "arn:aws:ec2:*::snapshot/*",
      "Condition": {
        "NumericGreaterThan": {
          "ec2:SnapshotTime": 48
        }
      }
    }
  ]
}
```

Mengunci snapshot Amazon EBS

Anda dapat mengunci snapshot yang ada dalam status `pending` atau `completed`. Untuk informasi selengkapnya, lihat Pertimbangan untuk kunci snapshot Amazon EBS.

Console

Untuk mengunci snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot yang akan dikunci dan pilih Tindakan, Pengaturan snapshot, Kelola kunci snapshot.
4. Pilih Kunci snapshot.
5. Untuk Mode kunci, pilih Mode tata kelola atau Mode kepatuhan. Untuk informasi selengkapnya, lihat Mode Kunci.
6. Untuk Durasi Penguncian, lakukan salah satu hal berikut:

   - Untuk mengunci snapshot untuk periode tertentu, pilih Kunci snapshot untuk, lalu masukkan periode dalam beberapa hari atau tahun.
   - Untuk mengunci snapshot hingga tanggal dan waktu tertentu, pilih Kunci snapshot hingga, lalu pilih tanggal dan waktu kedaluwarsa.

   Untuk informasi selengkapnya, lihat Durasi kunci.

7. (Hanya mode kepatuhan) Untuk Cooling-off periode, tentukan periode pendinginan di mana Anda dapat membuka kunci snapshot dan memodifikasi konfigurasi kunci. Untuk informasi selengkapnya, lihat Cooling-off periode.
8. (Hanya mode kepatuhan) Untuk mengonfirmasi bahwa Anda ingin mengunci snapshot dalam mode kepatuhan dan bahwa Anda tidak akan dapat membuka kunci snapshot setelah periode pendinginan berakhir, pilih Akui.
9. Pilih Simpan pengaturan kunci.

AWS CLI

Untuk mengunci snapshot dalam mode tata kelola

Gunakan perintah lock-snapshot. Untuk `--lock-mode`, tentukan `governance`. Untuk mengunci snapshot untuk periode tertentu, untuk`--lock-duration`, tentukan periode, dalam hari.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0abcdef1234567890 \
    --lock-mode governance \
    --lock-duration 30
```

Untuk mengunci snapshot hingga tanggal tertentu, untuk`--expiration-date`, tentukan tanggal dan waktu di mana kunci harus kedaluwarsa, di zona waktu UTC.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0abcdef1234567890 \
    --lock-mode governance \
    --expiration-date YYYY-MM-DDThh:mm:ss.sssZ
```

Untuk mengunci snapshot dalam mode kepatuhan

Gunakan perintah lock-snapshot. Untuk `--lock-mode`, tentukan `compliance`. Untuk`--cool-off-period`, secara opsional tentukan periode pendinginan, dalam jam. Untuk mengunci snapshot untuk periode tertentu, untuk`--lock-duration`, tentukan jumlah hari untuk mengunci snapshot.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0abcdef1234567890 \
    --lock-mode compliance \
    --cool-off-period 24 \
```

```
    --lock-duration 30
```

Untuk mengunci snapshot hingga tanggal tertentu, untuk`--expiration-date`, tentukan tanggal dan waktu di mana kunci harus kedaluwarsa, di zona waktu UTC.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0abcdef1234567890 \
    --lock-mode compliance \
    --expiration-date YYYY-MM-DDThh:mm:ss.sssZ
```

PowerShell

Untuk mengunci snapshot dalam mode tata kelola

Gunakan Lock-EC2Snapshot cmdlet. Anda dapat secara opsional menentukan durasi kunci snapshot, dalam hari.

```
Lock-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890 `
    -LockMode "governance" `
    -LockDuration 30
```

Atau, Anda dapat mengunci snapshot hingga tanggal tertentu, di zona waktu UTC.

```
Lock-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890 `
    -LockMode "governance" `
    -ExpirationDate YYYY-MM-DDThh:mm:ss.sssZ
```

Untuk mengunci snapshot dalam mode kepatuhan

Gunakan Lock-EC2Snapshot cmdlet. Anda dapat secara opsional menentukan periode pendinginan, dalam jam. Anda juga dapat secara opsional menentukan durasi kunci snapshot, dalam hari.

```
Lock-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890 `
    -LockMode "compliance" `
    -CoolOffPeriod 24 `
    -LockDuration 30
```

Atau, Anda dapat mengunci snapshot hingga tanggal tertentu, di zona waktu UTC.

```
Lock-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890 `
    -LockMode "compliance" `
    -ExpirationDate YYYY-MM-DDThh:mm:ss.sssZ
```

Membuka cuplikan snapshot Amazon EBS

Anda dapat membuka kunci snapshot hanya jika terkunci dalam mode tata kelola, atau jika terkunci dalam mode kepatuhan dan masih dalam periode pendinginan.

Console

Untuk membuka kunci snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot yang akan dibuka kuncinya dan pilih Tindakan, Pengaturan snapshot, Kelola kunci snapshot.
4. Pilih Buka kunci snapshot lalu pilih Buka kunci snapshot lagi untuk mengonfirmasi.

AWS CLI

Untuk membuka kunci snapshot

Gunakan perintah unlock-snapshot.

```
aws ec2 unlock-snapshot --snapshot-id snap-0abcdef1234567890
```

PowerShell

Untuk membuka kunci snapshot

Gunakan Unlock-EC2Snapshot cmdlet.

```
Unlock-EC2Snapshot -SnapshotId snap-0abcdef1234567890
```

Perbarui pengaturan kunci snapshot Amazon EBS

Pembaruan yang diizinkan tergantung pada status kunci:

- `governance` — Anda dapat mengubah mode kunci dan menambah atau mengurangi durasi kunci atau tanggal kedaluwarsa.
- `compliance-cooloff` — Anda dapat mengubah mode kunci, menambah atau mengurangi periode pendinginan, dan menambah atau mengurangi durasi kunci atau tanggal kedaluwarsa.
- `compliance` — Anda hanya dapat meningkatkan durasi kunci atau tanggal kedaluwarsa.

Console

Untuk memperbarui pengaturan kunci snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot yang pengaturan kuncinya akan dimodifikasi dan pilih Tindakan, Pengaturan snapshot, Kelola kunci snapshot.
4. Perbarui pengaturan sesuai kebutuhan, lalu pilih Simpan pengaturan kunci.

AWS CLI

Untuk memperbarui pengaturan kunci snapshot

Gunakan perintah lock-snapshot. Tentukan ID snapshot dan opsi untuk memodifikasi. Contoh berikut mengubah tanggal kedaluwarsa.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0abcdef1234567890 \
    --lock-mode governance \
    --expiration-date YYYY-MM-DDThh:mm:ss.sssZ
```

PowerShell

Untuk memperbarui pengaturan kunci snapshot

Gunakan Lock-EC2Snapshot cmdlet. Tentukan ID snapshot dan opsi untuk memodifikasi. Contoh berikut mengubah tanggal kedaluwarsa.

```
Lock-EC2Snapshot `
    -SnapshoId snap-0abcdef1234567890 `
    -LockMode "governance" `
    -ExpirationDate YYYY-MM-DDThh:mm:ss.sssZ
```

Pantau kunci snapshot Amazon EBS

Anda dapat memantau tindakan yang terkait dengan kunci snapshot Amazon EBS menggunakan alat berikut:

Topik

- Pantau kunci snapshot Amazon EBS menggunakan AWS CloudTrail
- Pantau kunci snapshot Amazon EBS menggunakan Amazon EventBridge

Pantau kunci snapshot Amazon EBS menggunakan AWS CloudTrail

Anda dapat memantau panggilan API untuk kunci snapshot sebagai peristiwa, termasuk panggilan dari konsol dan dari panggilan kode ke API. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan itu dibuat, dan detail tambahan.

Untuk informasi selengkapnya, lihat Log panggilan API menggunakan AWS CloudTrail.

Pantau kunci snapshot Amazon EBS menggunakan Amazon EventBridge

Amazon EBS memancarkan peristiwa yang terkait dengan tindakan kunci snapshot. Anda dapat menggunakan AWS Lambda dan Amazon EventBridge untuk menangani pemberitahuan acara secara terprogram. Peristiwa dipancarkan atas dasar upaya terbaik. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Peristiwa berikut dipancarkan:

- Snapshot berhasil dikunci dalam mode tata kelola atau kepatuhan.

```
{
  "version": "0",
  "id": "01234567-01234-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
```

```
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "lockSnapshot",
    "result": "succeeded",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef",
    "source": 012345678901,
    "lockState": "compliance-cooloff",
    "lockCreatedOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockExpiresOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockDuration": 123,
    "lockStartDurationTime": "yyyy-mm-ddThh:mm:ssZ",
    "cooOffPeriod": 24,
    "coolOffPeriodExpiresOn": "yyyy-mm-ddThh:mm:ssZ"
  }
}
```

- Peristiwa penguncian gagal jika snapshot dikunci saat berada dalam status `pending`, dan snapshot gagal mencapai status `completed`.

```
{
  "version": "0",
  "id": "01234567-01234-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "lockSnapshot",
    "result": "failed",
    "cause": "snapshot failed",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef",
    "lockState": "pending-compliance",
    "lockCreatedOn": "yyyy-mm-ddThh:mm:ssZ",
```

```
    "lockDuration": 123,
    "lockStartDurationTime": "yyyy-mm-ddThh:mm:ssZ",
    "cooOffPeriod": 24,
    "coolOffPeriodExpiresOn": "yyyy-mm-ddThh:mm:ssZ"
  }
}
```

- Kunci kedaluwarsa

```
{
  "version": "0",
  "id": "01234567-01234-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef"
  ],
  "detail": {
    "event": "lockDurationExpiry",
    "result": "succeeded",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef",
    "lockState": "expired",
    "lockCreatedOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockExpiresOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockDuration": 123
  }
}
```

- Cooling-off periode berakhir setelah dikunci dalam mode kepatuhan.

```
{
  "version": "0",
  "id": "01234567-01234-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef"
```

```
  ],
  "detail": {
    "event": "cooloffperiodExpiry",
    "result": "succeeded",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567890abcdef",
    "lockState": "compliance",
    "lockCreatedOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockExpiresOn": "yyyy-mm-ddThh:mm:ssZ",
    "lockDuration": 123,
    "lockStartDurationTime": "yyyy-mm-ddThh:mm:ssZ",
    "cooOffPeriod": 24,
    "coolOffPeriodExpiresOn": "yyyy-mm-ddThh:mm:ssZ"
  }
}
```

Blokir akses publik untuk snapshot Amazon EBS

Untuk mencegah berbagi snapshot secara publik, Anda sekarang dapat mengaktifkan blokir akses publik untuk snapshot. Setelah Anda mengaktifkan blokir akses publik untuk snapshot di Wilayah, setiap upaya untuk membagikan snapshot secara publik di Wilayah tersebut akan diblokir secara otomatis. Pengaktifan ini dapat membantu Anda meningkatkan keamanan snapshot dan untuk melindungi data snapshot Anda dari akses yang tidak terotorisasi atau tidak diinginkan.

Blokir akses publik untuk snapshot dapat diaktifkan dalam salah satu dari dua mode:

- Blokir semua pembagian — Memblokir semua pembagian snapshot Anda ke publik Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
- Blokir pembagian baru — Hanya memblokir pembagian snapshot baru ke publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.

Pertimbangan-pertimbangan

Ingatlah hal berikut saat bekerja dengan blokir akses publik untuk snapshot.

- Memblokir akses publik untuk snapshot tidak mencegah berbagi snapshot privat.
- Mengaktifkan akses publik blokir untuk snapshot dalam mode blokir semua berbagi tidak mengubah izin untuk snapshot yang sudah dibagikan secara publik. Sebaliknya, pengaktifan

ini mencegah snapshot agar tidak terlihat oleh publik dan dapat diakses publik. Oleh karena itu, atribut untuk snapshot ini masih menunjukkan bahwa snapshot tersebut dibagikan secara publik, meskipun tidak tersedia untuk umum.

Jika nanti Anda menonaktifkan blokir akses publik atau mengubah mode untuk memblokir berbagi baru, snapshot ini akan tersedia untuk umum lagi.

- Memblokir akses publik untuk snapshot adalah pengaturan Regional. Hal ini berlaku untuk semua snapshot di Wilayah tempatnya diaktifkan. Anda harus mengaktifkan blokir akses publik untuk snapshot di setiap Wilayah yang tidak Anda inginkan untuk membagi snapshot dengan publik.
- Blok publik akses adalah pengaturan tingkat akun. Ini berlaku untuk semua pengguna, termasuk pengguna administrator, di akun. Anda tidak dapat mengaktifkan blokir akses publik untuk snapshot di tingkat organisasi.
- Setelan blokir akses publik dikonfigurasi baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah setelan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.
- Memblokir akses publik untuk snapshot tidak mencegah berbagi EBS-backed AMI kepada publik. Jika Anda mengaktifkan blokir akses publik untuk snapshot, pengguna masih dapat berbagi EBS-backed AMI secara publik. Jika EBS-backed AMI dibagikan secara publik, pengguna dengan akses ke AMI tersebut dapat membuat volume dari snapshot terkait. Untuk mencegah pembagian publik dari AMI Anda, aktifkan blokir akses publik untuk AMI .
- Memblokir akses publik untuk snapshot tidak didukung dengan snapshot lokal aktif. AWS Outposts

Harga

Memblokir akses publik untuk snapshot dapat diaktifkan tanpa biaya tambahan.

Daftar Isi

- Izin IAM untuk memblokir akses publik untuk snapshot Amazon EBS
- Konfigurasikan akses publik blok untuk snapshot Amazon EBS
- Lihat pengaturan blokir akses publik untuk snapshot Amazon EBS
- Nonaktifkan blokir akses publik untuk snapshot Amazon EBS

- Pantau blokir akses publik untuk snapshot Amazon EBS menggunakan EventBridge

Izin IAM untuk memblokir akses publik untuk snapshot Amazon EBS

Secara default, pengguna tidak memiliki izin untuk bekerja dengan blokir akses publik untuk snapshot. 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.

Untuk bekerja dengan blok akses publik untuk snapshot, pengguna memerlukan izin berikut.

- `ec2:EnableSnapshotBlockPublicAccess` — Mengaktifkan blokir akses publik untuk snapshot dan mengubah mode.
- `ec2:DisableSnapshotBlockPublicAccess` — Menonaktifkan blokir akses publik untuk snapshot.
- `ec2:GetSnapshotBlockPublicAccessState` — Melihat blokir akses publik untuk pengaturan snapshot untuk Wilayah.

Berikut ini adalah contoh kebijakan IAM. Jika beberapa izin tidak diperlukan, Anda dapat menghapusnya dari kebijakan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "ec2:EnableSnapshotBlockPublicAccess",
            "ec2:DisableSnapshotBlockPublicAccess",
            "ec2:GetSnapshotBlockPublicAccessState"
        ],
        "Resource": "*"
    }]
}
```

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Konfigurasikan akses publik blok untuk snapshot Amazon EBS

Blokir akses publik untuk snapshot guna mencegah berbagi snapshot secara publik di Wilayah. Setelah fitur ini diaktifkan, permintaan untuk membagikan snapshot secara publik di Wilayah diblokir.

Important

Mengaktifkan akses publik blokir untuk snapshot dalam mode blokir semua berbagi tidak mengubah izin untuk snapshot yang sudah dibagikan secara publik. Sebaliknya, pengaktifan ini mencegah snapshot agar tidak terlihat oleh publik dan dapat diakses publik. Oleh karena itu, atribut untuk snapshot ini masih menunjukkan bahwa snapshot tersebut dibagikan secara publik, meskipun tidak tersedia untuk umum.
Jika nanti Anda menonaktifkan blokir akses publik atau mengubah mode untuk memblokir berbagi baru, snapshot ini akan tersedia untuk umum lagi.

Note

Pengaturan ini dikonfigurasi di tingkat akun, baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Itu harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin mencegah berbagi foto publik. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa

akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah setelan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Console

Untuk mengonfigurasi blokir akses publik untuk snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2, lalu di Atribut akun (di sisi kanan), pilih Perlindungan dan keamanan data.
3. Di bagian Blokir akses publik untuk snapshot EBS, pilih Kelola.
4. Pilih Blokir akses publik dan kemudian pilih salah satu opsi berikut:
   - Blokir semua akses publik — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
   - Blokir semua pembagian publik — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk umum.
5. Pilih Perbarui.

AWS CLI

Untuk mengaktifkan atau memodifikasi blokir akses publik untuk snapshot

Gunakan perintah enable-snapshot-block-public-access. Untuk `--state`, tentukan salah satu dari nilai-nilai berikut:

- `block-all-sharing` — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.

- `block-new-sharing` — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.

Untuk mengaktifkan atau memodifikasi akses publik blok untuk snapshot untuk Wilayah tertentu

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

Berikut ini adalah output contoh.

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

Untuk mengaktifkan atau memodifikasi akses publik blokir untuk snapshot untuk semua Wilayah

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 enable-snapshot-block-public-access \
            --region $region \
            --state block-new-sharing \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       block-new-sharing
```

```
eu-north-1       block-new-sharing
eu-west-3        block-new-sharing
...
```

PowerShell

Untuk mengaktifkan atau memodifikasi blokir akses publik untuk snapshot

Gunakan perintah Enable-EC2SnapshotBlockPublicAccess. Untuk `-State`, tentukan salah satu dari nilai-nilai berikut:

- `block-all-sharing` — Untuk memblokir semua pembagian snapshot untuk publik. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak lagi tersedia secara publik.
- `block-new-sharing` — Untuk memblokir hanya pembagian snapshot untuk publik baru. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.

Untuk mengaktifkan atau memodifikasi akses publik blok untuk snapshot untuk Wilayah tertentu

```
Enable-EC2SnapshotBlockPublicAccess `
    -Region us-east-1 `
    -State block-new-sharing
```

Berikut ini adalah output contoh.

```
Value
-----
block-new-sharing
```

Untuk mengaktifkan atau memodifikasi akses publik blokir untuk snapshot untuk semua Wilayah

```
(Get-EC2Region -Region us-east-1).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (
            Enable-EC2SnapshotBlockPublicAccess `
                -Region $_ `
```

```
                -State block-new-sharing)
    }
} | Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
Region         PublicAccessState
------         -----------------
ap-south-1     block-new-sharing
eu-north-1     block-new-sharing
eu-west-3      block-new-sharing
...
```

Lihat pengaturan blokir akses publik untuk snapshot Amazon EBS

Blokir akses publik dapat berada dalam salah satu status berikut untuk setiap Wilayah di akun Anda.

- Blokir semua pembagian — Semua pembagian snapshot untuk publik diblokir. Pengguna di akun tidak dapat meminta berbagi publik baru. Selain itu, snapshot yang sudah dibagikan secara publik diperlakukan sebagai privat dan tidak tersedia secara publik.
- Blokir pembagian baru — Hanya pembagian snapshot publik baru yang diblokir. Pengguna di akun tidak dapat meminta berbagi publik baru. Namun, snapshot yang sudah dibagikan secara publik, tetap tersedia untuk publik.
- Tidak diblokir — Pembagian publik tidak diblokir. Pengguna dapat berbagi snapshot secara publik.

Console

Untuk melihat pengaturan guna memblokir akses publik untuk snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2, lalu di Atribut akun (di sisi kanan), pilih Perlindungan dan keamanan data.
3. Bagian Blokir akses publik untuk snapshot EBS menunjukkan pengaturan saat ini.

AWS CLI

Untuk melihat pengaturan guna memblokir akses publik untuk snapshot

Gunakan perintah get-snapshot-block-public-access-state.

- Untuk Wilayah tertentu

```
aws ec2 get-snapshot-block-public-access-state
```

Dalam contoh output ini, `ManagedBy` bidang menunjukkan entitas yang mengonfigurasi pengaturan dan `account` menunjukkan bahwa pengaturan dikonfigurasi langsung di akun. Nilai berarti `declarative-policy` pengaturan dikonfigurasi oleh kebijakan deklaratif. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

```
{
    "State": "unblocked",
    "ManagedBy": "account"
}
```

- Untuk semua Wilayah

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 get-snapshot-block-public-access-state \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       unblocked
eu-north-1       unblocked
eu-west-3        unblocked
```

PowerShell

Untuk melihat pengaturan guna memblokir akses publik untuk snapshot

Gunakan Get-EC2SnapshotBlockPublicAccessState cmdlet.

- Untuk Wilayah tertentu

```
Get-EC2SnapshotBlockPublicAccessState -Region us-east-1
```

Berikut ini adalah output contoh.

```
Value
-----
block-new-sharing
```

- Untuk semua Wilayah

```
(Get-EC2Region -Region us-east-1).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (Get-EC2SnapshotBlockPublicAccessState -Region $_)
    }
} | Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       unblocked
eu-north-1       unblocked
eu-west-3        unblocked
...
```

Nonaktifkan blokir akses publik untuk snapshot Amazon EBS

Nonaktifkan blokir akses publik untuk snapshot guna mengizinkan berbagi snapshot secara publik. Setelah fitur ini dinonaktifkan, pengguna dapat membagikan snapshot secara publik di Wilayah.

Important

Mengaktifkan akses publik blokir untuk snapshot dalam mode blokir semua berbagi tidak mengubah izin untuk snapshot yang sudah dibagikan secara publik. Sebaliknya, pengaktifan ini mencegah snapshot agar tidak terlihat oleh publik dan dapat diakses publik. Oleh karena itu, atribut untuk snapshot ini masih menunjukkan bahwa snapshot tersebut dibagikan secara publik, meskipun tidak tersedia untuk umum.
Jika menonaktifkan blokir akses publik, snapshot ini akan tersedia untuk umum lagi.

Note

Pengaturan ini dikonfigurasi di tingkat akun, baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Itu harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin mengizinkan berbagi foto publik. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah setelan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Console

Untuk menonaktifkan blokir akses publik untuk snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2, lalu di Atribut akun (di sisi kanan), pilih Perlindungan dan keamanan data.
3. Di bagian Blokir akses publik untuk snapshot EBS, pilih Kelola.
4. Hapus Blokir akses publik dan pilih Perbarui.

AWS CLI

Untuk menonaktifkan blokir akses publik untuk snapshot

Gunakan perintah enable-snapshot-block-public-access.

- Untuk Wilayah tertentu

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

Berikut ini adalah output contoh.

```
{
    "State": "unblocked"
}
```

- Untuk semua Wilayah

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 disable-snapshot-block-public-access \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       unblocked
eu-north-1       unblocked
eu-west-3        unblocked
```

PowerShell

Untuk menonaktifkan blokir akses publik untuk snapshot

Gunakan Disable-EC2SnapshotBlockPublicAccess cmdlet.

- Untuk Wilayah tertentu

```
Disable-EC2SnapshotBlockPublicAccess -Region us-east-1
```

Berikut ini adalah output contoh.

```
Value
-----
unblocked
```

- Untuk semua Wilayah

```
(Get-EC2Region -Region us-east-1).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (Disable-EC2SnapshotBlockPublicAccess -Region $_)
    }
} | `
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
Region         PublicAccessState
------         -----------------
ap-south-1     unblocked
eu-north-1     unblocked
eu-west-3      unblocked
...
```

Pantau blokir akses publik untuk snapshot Amazon EBS menggunakan EventBridge

Amazon EBS memancarkan peristiwa terkait guna memblokir akses publik untuk snapshot. Anda dapat menggunakan AWS Lambda dan Amazon EventBridge untuk menangani pemberitahuan acara secara terprogram. Peristiwa dipancarkan atas dasar upaya terbaik. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Peristiwa berikut dipancarkan:

- Mengaktifkan blokir akses publik untuk snapshot dalam mode blokir semua pembagian

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Block Public Access Enabled",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2019-05-31T21:49:54Z",
  "region": "us-east-1",
  "detail": {
    "SnapshotBlockPublicAccessState": "block-all-sharing",
    "message": "Block Public Access was successfully enabled in 'block-all-sharing'
 mode"
  }
}
```

- Mengaktifkan blokir akses publik untuk snapshot dalam mode blokir pembagian baru

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Block Public Access Enabled",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "2019-05-31T21:49:54Z",
  "region": "us-east-1",
  "detail": {
    "SnapshotBlockPublicAccessState": "block-new-sharing",
    "message": "Block Public Access was successfully enabled in 'block-new-sharing'
 mode"
  }
}
```

- Menonaktifkan blokir akses publik untuk snapshot

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Block Public Access Disabled",
  "source": "aws.ec2",
```

```
  "account": "123456789012",
  "time": "2019-05-31T21:49:54Z",
  "region": "us-east-1",
  "detail": {
    "SnapshotBlockPublicAccessState": "unblocked",
    "message": "Block Public Access was successfully disabled"
  }
}
```

Snapshot lokal Amazon EBS di Outposts

Snapshot Amazon EBS adalah salinan titik waktu dari volume EBS Anda.

Secara default, snapshot volume EBS pada sebuah AWS Outpost disimpan di Amazon S3 di Wilayah. Outpost Anda juga dapat menggunakan snapshot lokal Amazon EBS di Outposts untuk menyimpan snapshot volume Outpost secara lokal di Amazon S3 pada itu sendiri. Outpost Ini memastikan bahwa data snapshot berada diOutpost, dan di tempat Anda. Selain itu, Anda dapat menggunakan kebijakan dan izin AWS Identity and Access Management (IAM) untuk mengatur kebijakan penegakan tempat tinggal data untuk memastikan bahwa data snapshot tidak keluar. Outpost Ini sangat berguna jika Anda tinggal di negara atau wilayah yang belum dilayani oleh Wil AWS ayah dan yang memiliki persyaratan residensi data.

Topik ini memberikan informasi tentang bekerja dengan Amazon EBS snapshot lokal di Outposts. Untuk informasi selengkapnya tentang snapshot Amazon EBS dan tentang bekerja dengan snapshot di suatu Wil AWS ayah, lihat. Snapshot Amazon EBS

Untuk informasi selengkapnya, lihat AWS Outposts Keluarga dan Dokumentasi AWS Outposts Keluarga.

Topik

- Pertanyaan umum
- Prasyarat
- Pertimbangan-pertimbangan
- Mengendalikan akses dengan IAM
- Bekerja dengan snapshot lokal

Pertanyaan umum

1. Apa itu snapshot lokal?

   Secara default, snapshot Amazon EBS dari volume pada an Outpost disimpan di Amazon S3 di Wilayah. Outpost Jika Outpost disediakan dengan S3 di Outposts, Anda dapat memilih untuk menyimpan snapshot secara lokal pada itu sendiri. Outpost Snapshot lokal bersifat inkremental, yang berarti hanya blok pada volume yang berubah setelah snapshot terbaru Anda disimpan. Anda dapat menggunakan snapshot ini untuk mengembalikan volume yang Outpost sama dengan snapshot kapan saja. Untuk informasi selengkapnya tentang snapshot Amazon EBS, lihat Snapshot Amazon EBS.

2. Mengapa saya harus menggunakan snapshot lokal?

   Snapshot adalah cara mudah untuk mencadangkan data Anda. Dengan snapshot lokal, semua data snapshot Anda disimpan secara lokal diOutpost. Ini berarti bahwa itu tidak meninggalkan tempat Anda. Ini sangat berguna jika Anda tinggal di negara atau wilayah yang belum dilayani oleh Wil AWS ayah dan yang memiliki persyaratan tempat tinggal.

   Selain itu, menggunakan snapshot lokal dapat membantu mengurangi bandwidth yang digunakan untuk komunikasi antara Wilayah dan Outpost lingkungan terbatas bandwidth.

3. Bagaimana cara menerapkan residensi data snapshot padaOutpost?

   Anda dapat menggunakan kebijakan AWS Identity and Access Management (IAM) untuk mengontrol izin yang dimiliki kepala sekolah (AWS akun, pengguna IAM, dan peran IAM) saat bekerja dengan snapshot lokal dan untuk menerapkan residensi data. Anda dapat membuat kebijakan yang mencegah kepala sekolah membuat snapshot dari Outpost volume dan instance dan menyimpan snapshot di Wilayah. AWS Saat ini, menyalin snapshot dan gambar dari suatu Outpost ke Wilayah tidak didukung. Untuk informasi selengkapnya, lihat Mengendalikan akses dengan IAM.

4. Apakah snapshot lokal multivolume dan crash-consistent didukung?

   Ya, Anda dapat membuat snapshot lokal multi-volume dan konsisten dengan crash dari instance pada. Outpost

5. Bagaimana cara membuat snapshot lokal?

   Anda dapat membuat snapshot secara manual menggunakan AWS Command Line Interface (AWS CLI) atau konsol Amazon EC2. Untuk informasi selengkapnya, lihat Bekerja dengan snapshot lokal. Anda juga dapat mengotomatiskan siklus hidup snapshot lokal menggunakan

Amazon Data Lifecycle Manager. Untuk informasi selengkapnya, lihat Mengotomatiskan snapshot pada Pos terdepan.

6. Dapatkah saya membuat, menggunakan, atau menghapus snapshot lokal jika saya Outpost kehilangan konektivitas ke Wilayahnya?

Tidak. OutpostHarus memiliki konektivitas dengan Wilayahnya karena Wilayah menyediakan akses, otorisasi, pencatatan, dan layanan pemantauan yang penting untuk kesehatan snapshot Anda. Jika tidak ada konektivitas, Anda tidak dapat membuat snapshot lokal baru, membuat volume atau meluncurkan instans dari snapshot lokal yang ada, atau menghapus snapshot lokal.

7. Seberapa cepat kapasitas penyimpanan Amazon S3 tersedia setelah menghapus snapshot lokal?

Kapasitas penyimpanan Amazon S3 tersedia dalam waktu 72 jam setelah menghapus snapshot lokal dan volume yang mereferensikannya.

8. Bagaimana saya bisa memastikan bahwa saya tidak kehabisan kapasitas Amazon S3 pada sayaOutpost?

Kami menyarankan Anda menggunakan CloudWatch alarm Amazon untuk memantau kapasitas penyimpanan Amazon S3 Anda, dan menghapus snapshot dan volume yang tidak lagi Anda perlukan untuk menghindari kehabisan kapasitas penyimpanan. Jika Anda menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan siklus hidup snapshot lokal, pastikan kebijakan penyimpanan snapshot Anda tidak menyimpan snapshot lebih lama dari yang diperlukan.

9. Apa yang terjadi jika saya kehabisan kapasitas Amazon S3 lokal pada sebuahOutpost?

Jika Anda kehabisan kapasitas Amazon S3 lokal pada sebuahOutpost, Amazon Data Lifecycle Manager tidak akan berhasil membuat snapshot lokal diOutpost. Amazon Data Lifecycle Manager akan mencoba membuat snapshot lokal diOutpost, tetapi snapshot segera beralih ke `error` status dan akhirnya dihapus oleh Amazon Data Lifecycle Manager. Sebaiknya gunakan met CloudWatch rik `SnapshotsCreateFailed` Amazon untuk memantau kebijakan siklus hidup snapshot untuk kegagalan pembuatan snapshot. Untuk informasi selengkapnya, lihat Memantau kebijakan Data Lifecycle Manager menggunakan CloudWatch.

10. Dapatkah saya menggunakan snapshot lokal dan AMI yang didukung oleh snapshot lokal dengan Instans Spot dan Armada Spot?

Tidak, Anda tidak dapat menggunakan snapshot lokal atau AMI yang didukung oleh snapshot lokal untuk meluncurkan Instans Spot atau Armada Spot.

11. Dapatkah saya menggunakan snapshot lokal dan AMI yang didukung oleh snapshot lokal dengan Amazon EC2 Auto Scaling?

Ya, Anda dapat menggunakan snapshot lokal dan AMI yang didukung oleh snapshot lokal untuk meluncurkan grup Penskalaan Otomatis di subnet yang Outpost sama dengan snapshot. Peran terkait layanan grup Amazon EC2 Auto Scaling harus memiliki izin untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi snapshot.

Anda tidak dapat menggunakan snapshot lokal atau AMI yang didukung oleh snapshot lokal untuk meluncurkan grup Penskalaan Otomatis di Wilayah. AWS

Prasyarat

Untuk menyimpan snapshot pada sebuahOutpost, Anda harus memiliki sebuah Outpost yang disediakan dengan S3 di Outposts. Untuk informasi selengkapnya tentang S3 di Pos Terdepan, lihat S3 di Pos Terdepan di Panduan Pengguna Amazon S3 on Outposts.

Pertimbangan-pertimbangan

Ingatlah hal-hal berikut ini saat bekerja dengan snapshot lokal.

- OutpostHarus memiliki konektivitas ke Wil AWS ayah mereka untuk menggunakan snapshot lokal.
- Metadata snapshot disimpan di Wil AWS ayah yang terkait denganOutpost. Hal ini tidak mencakup data snapshot.
- Snapshot yang disimpan di a di Outpost enkripsi secara default. Snapshot yang tidak dienkripsi tidak didukung. Saat Anda membuat volume pada snapshot Outpost lokal, volume dienkripsi menggunakan kunci KMS yang sama dengan snapshot sumber. Namun, Anda dapat mengubah kunci enkripsi saat menyalin snapshot lokal.
- Setelah Anda menghapus snapshot lokal dari sebuahOutpost, kapasitas penyimpanan Amazon S3 yang digunakan oleh snapshot yang dihapus akan tersedia dalam waktu 72 jam. Untuk informasi selengkapnya, lihat Menghapus snapshot lokal.
- Anda tidak dapat mengekspor snapshot lokal dari. Outpost
- Anda tidak dapat mengaktifkan pemulihan snapshot cepat untuk snapshot lokal.
- API langsung EBS tidak didukung dengan snapshot lokal.
- Anda tidak dapat menyalin snapshot lokal atau AMI dari satu Outpost ke Wil AWS ayah, dari satu Outpost ke yang lain, atau di dalam. Outpost Namun, Anda dapat menyalin snapshot dari Wil AWS

ayah ke. Outpost Untuk informasi selengkapnya, lihat Salin snapshot dari AWS Wilayah ke sebuah Pos terdepan.

- Saat menyalin snapshot dari Wil AWS ayah ke wilayahOutpost, data ditransfer melalui tautan layanan. Menyalin beberapa snapshot secara bersamaan dapat memengaruhi layanan lain yang berjalan diOutpost.
- Anda tidak dapat membagikan snapshot lokal.
- Anda harus menggunakan kebijakan IAM untuk memastikan bahwa persyaratan tempat tinggal data Anda terpenuhi. Untuk informasi selengkapnya, lihat Mengendalikan akses dengan IAM.
- Snapshot lokal bersifat cadangan inkremental. Hanya blok dalam volume yang telah berubah setelah snapshot terakhir yang disimpan. Setiap snapshot berisi semua informasi yang diperlukan untuk memulihkan data Anda (dari saat ketika snapshot diambil) ke volume EBS baru. Untuk informasi selengkapnya, lihat Cara kerja snapshot Amazon EBS.
- Anda tidak dapat menggunakan kebijakan IAM untuk menegakkan tempat tinggal data CopySnapshot dan CopyImage tindakan.

Mengendalikan akses dengan IAM

Anda dapat menggunakan kebijakan AWS Identity and Access Management (IAM) untuk mengontrol izin yang dimiliki kepala sekolah (AWS akun, pengguna IAM, dan peran IAM) saat bekerja dengan snapshot lokal. Berikut ini adalah contoh kebijakan yang dapat Anda gunakan untuk memberikan atau menolak izin untuk melakukan tindakan tertentu dengan snapshot lokal.

> **Important**
>
> Menyalin snapshot dan gambar dari suatu Outpost ke Wilayah saat ini tidak didukung. Akibatnya, saat ini Anda tidak dapat menggunakan kebijakan IAM untuk menerapkan residensi data untuk CopySnapshot dan CopyImage tindakan.

Topik

- Memberlakukan residensi data untuk snapshot
- Mencegah pengguna utama menghapus snapshot lokal

Memberlakukan residensi data untuk snapshot

Kebijakan contoh berikut mencegah semua prinsipal membuat snapshot dari volume dan instance Outpost `arn:aws:outposts:us-east-1:123456789012:outpost/op-1234567890abcdef` dan menyimpan data snapshot di Wilayah. AWS Pengguna utama masih dapat membuat snapshot lokal. Kebijakan ini memastikan bahwa semua snapshot tetap berada diOutpost.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots"
            ],
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "ec2:SourceOutpostArn": "arn:aws:outposts:us-
east-1:123456789012:outpost/op-1234567890abcdef0"
                },
                "Null": {
                    "ec2:OutpostArn": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots"
            ],
            "Resource": "*"
        }
    ]
}
```

Mencegah pengguna utama menghapus snapshot lokal

Kebijakan contoh berikut mencegah semua prinsipal menghapus snapshot lokal yang disimpan di. Outpost `arn:aws:outposts:us-east-1:123456789012:outpost/op-1234567890abcdef0`

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ec2:DeleteSnapshot"
            ],
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "ec2:OutpostArn": "arn:aws:outposts:us-
east-1:123456789012:outpost/op-1234567890abcdef0"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteSnapshot"
            ],
            "Resource": "*"
        }
    ]
}
```

Bekerja dengan snapshot lokal

Bagian berikut menjelaskan cara menggunakan snapshot lokal.

Topik

- Aturan untuk menyimpan snapshot
- Buat snapshot lokal dari volume pada Pos terdepan

- Membuat AMI dari snapshot lokal
- Salin snapshot dari AWS Wilayah ke sebuah Pos terdepan
- Menyalin AMI dari AWS Wilayah ke sebuah Pos terdepan
- Membuat volume dari snapshot lokal
- Meluncurkan instans dari AMI didukung oleh snapshot lokal
- Menghapus snapshot lokal
- Mengotomatiskan snapshot pada Pos terdepan

Aturan untuk menyimpan snapshot

Aturan berikut berlaku untuk penyimpanan snapshot:

- Jika snapshot terbaru dari volume disimpan pada sebuahOutpost, maka semua snapshot berturut-turut harus disimpan pada volume yang samaOutpost.
- Jika snapshot terbaru dari volume disimpan di Wil AWS ayah, maka semua snapshot berturut-turut harus disimpan di Wilayah yang sama. Untuk mulai membuat snapshot lokal dari volume tersebut, lakukan hal berikut:

  1. Buat snapshot volume di Wil AWS ayah.
  2. Salin snapshot ke Outpost dari Wil AWS ayah.
  3. Buat volume baru dari snapshot lokal.
  4. Pasang volume ke instance diOutpost.

  Untuk volume baru diOutpost, snapshot berikutnya dapat disimpan di Outpost atau di Wil AWS ayah. Semua snapshot berturut-turut kemudian harus disimpan di lokasi yang sama.
- Snapshot lokal, termasuk snapshot yang dibuat pada Outpost dan snapshot yang disalin ke Outpost dari Wil AWS ayah, hanya dapat digunakan untuk membuat volume yang sama. Outpost
- Jika Anda membuat volume pada Outpost dari snapshot di Wilayah, maka semua snapshot berturut-turut dari volume baru itu harus berada di Wilayah yang sama.
- Jika Anda membuat volume pada snapshot lokal, maka semua snapshot berturut-turut dari volume baru itu harus samaOutpost. Outpost

Buat snapshot lokal dari volume pada Pos terdepan

Anda dapat membuat snapshot lokal dari volume di. Outpost Anda dapat memilih untuk menyimpan snapshot pada volume sumber yang Outpost sama, atau di Wilayah untuk. Outpost

Snapshot lokal dapat digunakan untuk membuat volume pada yang sama Outpost saja.

Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS

Membuat AMI dari snapshot lokal

Anda dapat membuat Amazon Machine Images (AMI) menggunakan kombinasi snapshot lokal dan snapshot yang disimpan di Wilayah. Outpost Misalnya, jika Anda memiliki Outpost in`us-east-1`, Anda dapat membuat AMI dengan volume data yang didukung oleh snapshot lokal pada ituOutpost, dan volume root yang didukung oleh snapshot di Wil `us-east-1` ayah.

Anda juga dapat membuat AMI langsung dari instance Outpost yang mendukung snapshot lokal. Snapshot cadangan yang dibuat untuk AMI dapat disimpan Outpost sama dengan instance atau di Wilayah induknya.

> **Note**
>
> - Anda tidak dapat membuat AMI yang menyertakan snapshot cadangan yang disimpan di beberapaOutposts.
> - AMI yang didukung oleh snapshot lokal dapat digunakan untuk meluncurkan instans pada instans yang sama Outpost saja.

Untuk membuat AMI pada Pos terdepan dari snapshot di suatu Wilayah

1. Salin snapshot dari Wilayah keOutpost. Untuk informasi selengkapnya, lihat Salin snapshot dari AWS Wilayah ke sebuah Pos terdepan.
2. Gunakan konsol Amazon EC2 atau perintah register-image untuk membuat AMI menggunakan salinan snapshot di. Outpost Untuk informasi selengkapnya, lihat Membuat AMI dari suatu snapshot.

Untuk membuat AMI secara langsung dari instance pada Pos terdepan

Gunakan perintah create-image dengan `--snapshot-location` parameter untuk mengontrol tempat penyimpanan snapshot pendukung:

- Untuk menyimpan snapshot secara lokal Outpost sama dengan instance, tent `local` ukan.
- Untuk menyimpan snapshot di Wilayah indukOutpost, tent `regional` ukan.

Jika instance sumber ada pada Outpost yang mendukung snapshot lokal, `--snapshot-location` parameter diperlukan. Jika Anda menghilangkannya, permintaan gagal dengan `InvalidParameterValue` kesalahan.

Contoh berikut membuat AMI dan menyimpan snapshot cadangan pada contoh yang Outpost sama dengan instance.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "My Outpost image" \
    --snapshot-location local
```

Untuk membuat AMI di Wilayah dari instance pada Pos terdepan

Gunakan perintah create-image dan tentukan `regional` untuk parameter. `--snapshot-location`

Contoh berikut membuat AMI dan menyimpan snapshot pendukung di Wilayah indukOutpost.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "My Regional image" \
    --snapshot-location regional
```

Salin snapshot dari AWS Wilayah ke sebuah Pos terdepan

Anda dapat menyalin snapshot dari Wil AWS ayah ke. Outpost Anda dapat melakukan ini hanya jika snapshot berada di Wilayah untuk. Outpost Jika snapshot berada di Wilayah yang berbeda, Anda harus terlebih dahulu menyalin snapshot ke Wilayah untukOutpost, dan kemudian menyalinnya dari Wilayah tersebut Outpost ke.

> **Note**
>
> Anda tidak dapat menyalin snapshot lokal dari satu Outpost ke Wilayah, dari satu Outpost ke yang lain, atau dalam yang samaOutpost.

Untuk informasi selengkapnya, lihat Menyalin snapshot Amazon EBS.

Menyalin AMI dari AWS Wilayah ke sebuah Pos terdepan

Anda dapat menyalin AMI dari AWS Wilayah keOutpost. Saat Anda menyalin AMI dari Wilayah ke sebuahOutpost, semua snapshot yang terkait dengan AMI disalin dari Wilayah keOutpost.

Anda dapat menyalin AMI dari Wilayah ke wilayah Outpost hanya jika snapshot yang terkait dengan AMI berada di Wilayah untuk. Outpost Jika snapshot berada di Wilayah yang berbeda, Anda harus terlebih dahulu menyalin AMI ke Wilayah untukOutpost, dan kemudian menyalinnya dari Wilayah tersebut Outpost ke.

> **Note**
>
> Anda tidak dapat menyalin AMI dari satu Outpost ke Wilayah, dari satu Outpost ke yang lain, atau di dalamOutpost.

Anda dapat menyalin AMI dari Wilayah ke sebuah Outpost menggunakan AWS CLI perintah copy-image saja.

Membuat volume dari snapshot lokal

Anda dapat membuat volume pada Outpost snapshot lokal. Volume harus dibuat Outpost sama dengan snapshot sumber. Anda tidak dapat menggunakan snapshot lokal untuk membuat volume di Wilayah untuk. Outpost

Ketika Anda membuat volume dari snapshot lokal, Anda tidak dapat mengenkripsi ulang volume menggunakan kunci KMS yang berbeda. Volume yang dibuat dari snapshot lokal harus dienkripsi menggunakan kunci KMS yang sama dengan snapshot sumber.

Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.

Meluncurkan instans dari AMI didukung oleh snapshot lokal

Anda dapat meluncurkan instans dari AMI didukung oleh snapshot lokal Anda harus meluncurkan Instans Outpost sama dengan AMI sumber. Untuk informasi selengkapnya, lihat Mel uncurkan instance pada Anda Outpost di Panduan AWS Outposts Pengguna.

Menghapus snapshot lokal

Anda dapat menghapus snapshot lokal dari sebuahOutpost. Setelah Anda menghapus snapshot dari sebuahOutpost, kapasitas penyimpanan Amazon S3 yang digunakan oleh snapshot yang dihapus akan tersedia dalam waktu 72 jam setelah menghapus snapshot dan volume yang mereferensikan snapshot tersebut.

Karena kapasitas penyimpanan Amazon S3 tidak segera tersedia, kami sarankan Anda menggunakan CloudWatch alarm Amazon untuk memantau kapasitas penyimpanan Amazon S3 Anda. Hapus snapshot dan volume yang tidak Anda perlukan lagi untuk menghindari kehabisan kapasitas penyimpanan.

Untuk informasi selengkapnya tentang menghapus snapshot, lihat Menghapus snapshot.

Mengotomatiskan snapshot pada Pos terdepan

Anda dapat membuat kebijakan siklus hidup snapshot Amazon Data Lifecycle Manager yang secara otomatis membuat, menyalin, menyimpan, dan menghapus snapshot volume dan instans Anda pada. Outpost Anda dapat memilih apakah akan menyimpan snapshot di Wilayah atau apakah akan menyimpannya secara lokal di. Outpost Selain itu, Anda dapat secara otomatis menyalin snapshot yang dibuat dan disimpan di Wil AWS ayah ke. Outpost

Tabel berikut memberikan gambaran umum fitur yang didukung.

Lokasi sumber daya	Tujuan snapshot	Cross-region salin ke Wilayah	Cross-region salin ke Outpost	Pemulihan snapshot cepat	Cross-account berbagi
Region	Region	✓	✓	✓	✓
Outpost	Region	✓	✓	✓	✓
Outpost	Outpost	✗	✗	✗	✗

Pertimbangan-pertimbangan

- Hanya kebijakan siklus hidup snapshot Amazon EBS yang saat ini didukung. EBS-backed Kebijakan AMI dan kebijakan acara Cross-account berbagi tidak didukung.
- Jika kebijakan mengelola snapshot untuk volume atau instans di suatu Wilayah, snapshot dibuat di Wilayah yang sama dengan sumber daya sumber.
- Jika kebijakan mengelola snapshot untuk volume atau instance pada sebuahOutpost, maka snapshot dapat dibuat pada sumberOutpost, atau di Wilayah untuk itu. Outpost
- Kebijakan tunggal tidak dapat mengelola snapshot di Wilayah dan snapshot pada suatu. Outpost Jika Anda perlu mengotomatiskan snapshot di Wilayah dan di suatuOutpost, Anda harus membuat kebijakan terpisah.
- Pemulihan snapshot cepat tidak didukung untuk snapshot yang dibuat padaOutpost, atau untuk snapshot yang disalin ke. Outpost
- Cross-account berbagi tidak didukung untuk snapshot yang Outpost dibuat pada.
- Pre-scripts dan post-script tidak didukung untuk snapshot yang dibuat pada file. Outpost

Untuk informasi selengkapnya tentang membuat siklus hidup snapshot yang mengelola snapshot lokal, lihat Mengotomatisasi siklus hidup snapshot.

Snapshot lokal di Zona Lokal

Snapshot Amazon EBS adalah salinan titik waktu dari volume EBS Anda.

Snapshot volume EBS di Zona AWS Lokal dapat disimpan di Amazon S3 di Zona Lokal yang sama atau di Wilayah induk Zona Lokal tersebut. Menyimpan snapshot di Zona Lokal dapat membantu Anda memenuhi kebutuhan residensi data dengan memastikan bahwa data snapshot diproses dan disimpan di negara, negara bagian, atau kotamadya tertentu. Anda juga dapat mengatur kebijakan penegakan residensi data menggunakan IAM untuk memastikan bahwa data snapshot tidak meninggalkan Zona Lokal.

Zona Lokal ideal untuk aplikasi yang memerlukan latensi milidetik satu digit atau pemrosesan data lokal dengan membawa AWS infrastruktur lebih dekat ke pengguna akhir dan pusat bisnis Anda. Selain itu, Anda dapat memenuhi persyaratan residensi data untuk beban kerja yang sensitif terhadap peraturan dan kepatuhan. Untuk informasi selengkapnya, lihat Apa itu AWS Zona Lokal.

Snapshot lokal saat ini didukung di Zona Lokal yang mendukung Amazon S3. Untuk informasi selengkapnya, lihat Fitur Zona AWS Lokal. .

Topik

- Pertanyaan umum
- Pertimbangan-pertimbangan
- Mengendalikan akses dengan IAM

Pertanyaan umum

1. Apa itu snapshot Lokal di Zona Lokal?

   Snapshot lokal di Zona Lokal adalah snapshot yang disimpan di Amazon S3 di Zona Lokal. Secara default, snapshot volume Amazon EBS di Zona Lokal disimpan di Amazon S3 di Wilayah induk. Jika Zona Lokal mendukung Amazon S3, Anda dapat memilih untuk menyimpan snapshot secara lokal di Zona Lokal sebagai gantinya. Seperti snapshot di Wil AWS ayah, snapshot Lokal di Zona Lokal bersifat inkremental, yang berarti bahwa hanya blok volume yang telah berubah setelah snapshot terbaru Anda disimpan. Anda dapat menggunakan snapshot ini untuk memulihkan volume Amazon EBS di Zona Lokal yang sama kapan saja.

2. Mengapa saya harus menggunakan snapshot lokal?

   Gunakan snapshot Lokal di Zona Lokal untuk memenuhi persyaratan residensi data atau isolasi data dengan memastikan bahwa data snapshot Anda berada di lokasi geografis tertentu, seperti negara, negara bagian, atau kotamadya.

3. Bagaimana cara menerapkan residensi data snapshot di Zona Lokal?

   Anda dapat menggunakan kebijakan AWS Identity and Access Management (IAM) untuk mengontrol izin yang dimiliki kepala sekolah (AWS akun, pengguna IAM, dan peran IAM) saat bekerja dengan snapshot Lokal di Zona Lokal dan untuk menerapkan residensi data. Misalnya, Anda dapat membuat kebijakan yang mencegah pengguna membuat snapshot dari volume di Zona Lokal dan menyimpan snapshot tersebut di Wilayah. AWS Untuk informasi selengkapnya, lihat Mengendalikan akses dengan IAM.

4. Apakah snapshot Lokal multi-volume dan konsisten terhadap crash didukung?

   Ya, Anda dapat membuat snapshot Lokal multi-volume dan konsisten crash di Zona Lokal dari instance di Zona Lokal.

5. Bagaimana cara membuat snapshot lokal di Zona Lokal?

   Anda dapat membuat snapshot Lokal di Zona Lokal secara manual menggunakan AWS CLI atau konsol Amazon EC2. Untuk informasi selengkapnya, lihat Membuat snapshot volume EBS. Anda

juga dapat mengotomatiskan siklus hidup snapshot Lokal di Zona Lokal menggunakan Amazon Data Lifecycle Manager. Untuk informasi selengkapnya, lihat Buat kebijakan kustom Amazon Data Lifecycle Manager untuk snapshot EBS.

6. Bisakah saya menyalin snapshot Lokal di Zona Lokal?

   Ya, Anda dapat menyalin snapshot dari Wilayah ke Zona Lokal di Wilayah yang sama, dari Zona Lokal ke Wilayahnya, dan dari satu Zona Lokal ke Zona Lokal lainnya dalam grup Zona Lokal yang sama.

7. Bagaimana cara memulihkan data dari snapshot Lokal di Zona Lokal?

   Anda dapat menggunakan snapshot Lokal di Zona Lokal untuk membuat volume Amazon EBS hanya di Zona Lokal yang sama.

8. Bagaimana snapshot Lokal di Zona Lokal dienkripsi?

   Lihat Enkripsi Amazon EBS di Zona Lokal, Pos Terdepan, dan Zona Panjang Gelombang untuk perilaku Zone-specific enkripsi lokal. Di Zona Lokal di mana sumber daya Amazon EBS dienkripsi secara default, volume yang dibuat dari snapshot lokal selalu dienkripsi menggunakan AWS KMS kunci yang sama dengan snapshot. Namun, Anda dapat mengubah kunci enkripsi saat menyalin snapshot lokal.

9. Dapatkah saya EBS-backed membuat AMI menggunakan snapshot Lokal di Zona Lokal?

   Ya, Anda dapat menggunakan snapshot Lokal di Zona Lokal saat membuat EBS-backed AMI dengan menentukan tujuan snapshot sebagai Zona Lokal. Untuk informasi selengkapnya, lihat Membuat EBS-backed AMI Amazon.

10. Dapatkah saya membagikan snapshot Lokal di Zona Lokal?

    Ya, Anda dapat berbagi snapshot Lokal di Zona Lokal dengan AWS akun lain yang telah mengaktifkan Zona Lokal untuk digunakan di akun mereka.

11. Dapatkah saya membuat snapshot lokal dari volume dan kemudian beralih ke membuat snapshot di Wilayah induk?

    Tidak, setelah Anda membuat snapshot lokal dari volume, Anda tidak dapat membuat snapshot berturut-turut dari volume tersebut di Wilayah Induk. Karena semua snapshot bersifat inkremental, jika snapshot terbaru dari volume adalah snapshot Lokal, maka semua snapshot berturut-turut dari volume tersebut harus berupa snapshot lokal.

Pertimbangan-pertimbangan

Ingatlah hal berikut saat bekerja dengan snapshot Lokal di Zona Lokal.

- Snapshot lokal saat ini didukung di Zona Lokal yang mendukung Amazon S3.
- Fitur berikut tidak dapat digunakan dengan snapshot Lokal di Zona Lokal:
  - Tindakan Import/Export VM
  - Pemulihan snapshot cepat
  - API langsung EBS
  - Keranjang Daur Ulang
  - Arsip snapshot
- Anda harus menggunakan kebijakan IAM untuk menegakkan persyaratan tempat tinggal data Anda. Untuk informasi selengkapnya, lihat Mengendalikan akses dengan IAM.
- Jika snapshot terbaru dari volume adalah snapshot lokal, maka semua snapshot berturut-turut harus berupa snapshot lokal. Demikian pula, jika snapshot terbaru dari volume disimpan di Wil AWS ayah, maka semua snapshot berturut-turut harus disimpan di Wilayah yang sama.

Mengendalikan akses dengan IAM

Anda dapat menggunakan kebijakan AWS Identity and Access Management (IAM) untuk mengontrol izin yang dimiliki kepala sekolah (AWS akun, pengguna IAM, dan peran IAM) saat bekerja dengan snapshot Lokal di Zona Lokal. Berikut ini adalah contoh kebijakan yang dapat Anda gunakan untuk memberikan atau menolak izin untuk melakukan tindakan tertentu dengan snapshot Lokal di Zona Lokal.

Topik

- Menerapkan residensi data untuk snapshot Lokal di Zona Lokal
- Mencegah berbagi snapshot Lokal di Zona Lokal
- Mencegah kepala sekolah menghapus snapshot Lokal di Zona Lokal

Menerapkan residensi data untuk snapshot Lokal di Zona Lokal

Kebijakan contoh berikut membatasi pengguna untuk hanya membuat snapshot Lokal di Zona Lokal dari volume dan instance di Zona Lokal. Ini mencegah pengguna membuat snapshot di Wilayah dari volume dan instance di Zona Lokal.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":
    [
        {
            "Effect": "Deny",
            "Action":
            [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots"
            ],
            "Resource": "*",
            "Condition":
            {
                "StringEquals":
                {
                    "ec2:Location": "regional",
                    "ec2:SourceAvailabilityZone": "local_zone"
                }
            }
        },
        {
            "Effect": "Deny",
            "Action":
            [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots"
            ],
            "Resource": "*",
            "Condition":
            {
                "StringEquals":
                {
                    "ec2:SourceAvailabilityZone": "local_zone"
                },
                "Null":
                {
                    "ec2:Location": "true"
                }
            }
        },
```

```
        {
            "Effect": "Allow",
            "Action":
            [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots"
            ],
            "Resource": "*"
        }
    ]
}
```

Mencegah berbagi snapshot Lokal di Zona Lokal

Kebijakan contoh berikut mencegah semua pengguna berbagi snapshot Lokal di Zona Lokal.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "DenySnapshotModifyInLocalZone",
      "Effect": "Deny",
      "Action": [
        "ec2:ModifySnapshotAttribute"
      ],
      "Resource": "arn:aws:ec2:*::snapshot/*",
      "Condition": {
        "StringEquals": {
        "ec2:AvailabilityZone": "use1-atl2-az1"
        }
      }
    },
    {
      "Sid": "AllowSnapshotModifyElsewhere",
      "Effect": "Allow",
      "Action": [
        "ec2:ModifySnapshotAttribute"
      ],
      "Resource": "arn:aws:ec2:*::snapshot/*"
```

```
    }
  ]
}
```

Mencegah kepala sekolah menghapus snapshot Lokal di Zona Lokal

Kebijakan contoh berikut mencegah semua pengguna menghapus snapshot Lokal di Zona Lokal.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "DenySnapshotDeleteInLocalZone",
      "Effect": "Deny",
      "Action": [
        "ec2:DeleteSnapshot"
      ],
      "Resource": "arn:aws:ec2:*::snapshot/*",
      "Condition": {
        "StringEquals": {
        "ec2:AvailabilityZone": "use1-atl2-az1"
        }
      }
    },
    {
      "Sid": "AllowSnapshotDeleteElsewhere",
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteSnapshot"
      ],
      "Resource": "arn:aws:ec2:*::snapshot/*"
    }
  ]
}
```

Enkripsi EBS Amazon

Gunakan enkripsi Amazon EBS sebagai solusi enkripsi langsung untuk sumber daya Amazon EBS yang terkait dengan instans Amazon EC2 Anda. Dengan enkripsi Amazon EBS, Anda tidak perlu membangun, memelihara, dan mengamankan infrastruktur manajemen kunci Anda sendiri. Enkripsi Amazon EBS menggunakan AWS KMS keys saat membuat volume dan snapshot yang terenkripsi.

Operasi enkripsi terjadi pada server yang melakukan hosting instans EC2, memastikan keamanan data diam dan data bergerak antara instans serta penyimpanan EBS yang terlampir.

Anda dapat melampirkan volume terenkripsi maupun tak terenkripsi ke suatu instans secara bersamaan. Semua jenis instans Amazon EC2 mendukung enkripsi Amazon EBS.

Daftar Isi

- Cara kerja enkripsi Amazon EBS
- Enkripsi Amazon EBS di Zona Lokal, Pos Terdepan, dan Zona Panjang Gelombang
- Persyaratan untuk enkripsi Amazon EBS
- Aktifkan enkripsi Amazon EBS secara default
- Enkripsi sumber daya EBS
- Putar AWS KMS kunci yang digunakan untuk enkripsi Amazon EBS
- Contoh enkripsi Amazon EBS

Cara kerja enkripsi Amazon EBS

Anda dapat mengenkripsi volume boot dan data dari instans EC2.

Saat Anda membuat volume EBS terenkripsi dan melampirkannya ke tipe instans yang didukung, tipe data berikut dienkripsi:

- Data diam di dalam volume
- Semua data yang bergerak antara volume dan instans
- Semua snapshot yang dibuat dari volume
- Semua volume yang dibuat dari snapshot tersebut

Amazon EBS mengenkripsi volume Anda dengan kunci data menggunakan enkripsi data standar industri AES-256 . Kunci data dihasilkan oleh AWS KMS dan kemudian dienkripsi AWS KMS dengan AWS KMS kunci sebelum disimpan dengan informasi volume Anda. Amazon EBS secara otomatis membuat unik Kunci yang dikelola AWS di setiap Wilayah tempat Anda membuat sumber daya Amazon EBS. Alias untuk kunci KMS adalah`aws/ebs`. Secara default, Amazon EBS menggunakan kunci KMS ini untuk enkripsi. Atau, Anda dapat menggunakan kunci enkripsi yang dikelola pelanggan simetris yang Anda buat. Penggunaan kunci KMS sendiri akan memberikan Anda fleksibilitas yang lebih baik, termasuk kemampuan untuk membuat, memutar, dan menonaktifkan kunci KMS.

Amazon EC2 bekerja dengan AWS KMS untuk mengenkripsi dan mendekripsi volume EBS Anda dengan cara yang sedikit berbeda tergantung pada apakah snapshot tempat Anda membuat volume terenkripsi dienkripsi atau tidak.

Cara kerja enkripsi EBS saat snapshot dienkripsi

Saat Anda membuat volume terenkripsi dari snapshot terenkripsi yang Anda miliki, Amazon EC2 bekerja dengan AWS KMS untuk mengenkripsi dan mendekripsi volume EBS Anda sebagai berikut:

1. Amazon EC2 mengirimkan GenerateDataKeyWithoutPlaintext permintaan ke AWS KMS, menentukan kunci KMS yang Anda pilih untuk enkripsi volume.
2. Jika volume dienkripsi menggunakan kunci KMS yang sama dengan snapshot, AWS KMS gunakan kunci data yang sama dengan snapshot dan mengenkripsinya di bawah kunci KMS yang sama. Jika volume dienkripsi menggunakan kunci KMS yang berbeda, AWS KMS menghasilkan kunci data baru dan mengenkripsinya di bawah kunci KMS yang Anda tentukan. Kunci data terenkripsi dikirimkan ke Amazon EBS untuk disimpan dengan metadata volume.
3. Saat Anda melampirkan volume terenkripsi ke instans, Amazon EC2 mengirimkan CreateGrant permintaan AWS KMS agar dapat mendekripsi kunci data.
4. AWS KMS mendekripsi kunci data terenkripsi dan mengirimkan kunci data yang didekripsi ke Amazon EC2.
5. Amazon EC2 menggunakan kunci data plaintext di perangkat keras Nitro untuk mengenkripsi disk I/O ke volume. Kunci data teks biasa tetap ada di memori selama volumenya dipasang pada instans.

Cara kerja enkripsi EBS saat snapshot yang tidak terenkripsi

Ketika Anda membuat volume terenkripsi dari snapshot yang tidak terenkripsi yang Anda miliki, Amazon EC2 bekerja dengan AWS KMS untuk mengenkripsi dan mendekripsi volume EBS Anda sebagai berikut:

1. Amazon EC2 mengirimkan CreateGrant permintaan ke AWS KMS, sehingga dapat mengenkripsi volume yang dibuat dari snapshot.
2. Amazon EC2 mengirimkan GenerateDataKeyWithoutPlaintext permintaan ke AWS KMS, menentukan kunci KMS yang Anda pilih untuk enkripsi volume.
3. AWS KMS menghasilkan kunci data baru, mengenkripsinya di bawah kunci KMS yang Anda pilih untuk enkripsi volume, dan mengirimkan kunci data terenkripsi ke Amazon EBS untuk disimpan dengan metadata volume.
4. Amazon EC2 mengirimkan permintaan Decrypt AWS KMS untuk mendekripsi kunci data terenkripsi, yang kemudian digunakan untuk mengenkripsi data volume.
5. Saat Anda melampirkan volume terenkripsi ke instans, Amazon EC2 mengirimkan CreateGrant permintaan ke AWS KMS, sehingga dapat mendekripsi kunci data.
6. Saat Anda melampirkan volume terenkripsi ke instans, Amazon EC2 mengirimkan permintaan Dec rypt ke AWS KMS, menentukan kunci data terenkripsi.
7. AWS KMS mendekripsi kunci data terenkripsi dan mengirimkan kunci data yang didekripsi ke Amazon EC2.
8. Amazon EC2 menggunakan kunci data plaintext di perangkat keras Nitro untuk mengenkripsi disk I/O ke volume. Kunci data teks biasa tetap ada di memori selama volumenya dilampirkan pada instans.

Untuk informasi selengkapnya, lihat Cara Amazon Elastic Block Store (Amazon EBS) menggunakan AWS KMS dan Contoh dua Amazon EC2 dalam Panduan Developer AWS Key Management Service .

Bagaimana kunci KMS yang tidak dapat digunakan memengaruhi kunci data

Ketika kunci KMS menjadi tidak dapat digunakan, efeknya hampir seketika (tergantung pada konsistensi akhirnya). Status kunci dari perubahan kunci KMS untuk mencerminkan kondisi barunya, dan semua permintaan untuk menggunakan kunci KMS dalam operasi kriptografi gagal.

Saat Anda melakukan tindakan yang membuat kunci KMS tidak dapat digunakan, tidak akan ada efek langsung pada instans EC2 atau volume EBS yang dilampirkan. Amazon EC2 menggunakan kunci data, bukan kunci KMS, untuk mengenkripsi semua disk I/O saat volume terpasang ke instans.

Namun, saat volume EBS terenkripsi dicopot dari instans EC2, Amazon EBS menghapus kunci data dari perangkat keras Nitro. Pada saat volume EBS yang terenkripsi dilampirkan ke instans EC2, pelampiran akan gagal karena Amazon EBS tidak dapat menggunakan kunci KMS untuk mendekripsi kunci data terenkripsi dari volume tersebut. Untuk menggunakan volume EBS lagi, Anda harus membuat kunci KMS dapat digunakan lagi.

> **Tip**
>
> Jika Anda tidak lagi menginginkan akses ke data yang disimpan dalam volume EBS yang dienkripsi dengan kunci data yang dihasilkan dari kunci KMS yang ingin Anda buat agar tidak dapat digunakan, sebaiknya copot volume EBS dari instans EC2 sebelum Anda membuat kunci KMS tidak dapat digunakan.

Untuk informasi selengkapnya, lihat Bagaimana kunci KMS yang tidak dapat digunakan memengaruhi kunci data di Panduan Developer AWS Key Management Service .

Enkripsi Amazon EBS di Zona Lokal, Pos Terdepan, dan Zona Panjang Gelombang

Perilaku enkripsi Amazon EBS bervariasi tergantung pada jenis lokasi tempat Anda membuat sumber daya.

Zona Lokal

Di Zona Lokal berikut, semua sumber daya Amazon EBS dienkripsi secara default. Anda tidak dapat membuat volume atau snapshot yang tidak dienkripsi. Jika Anda tidak menentukan kunci KMS, Amazon EBS mengenkripsi sumber daya Anda dengan kunci enkripsi default akun Anda. Secara default, ini adalah Kunci yang dikelola AWS (`aws/ebs`). Untuk mengubah kunci enkripsi default untuk akun Anda, lihatKunci enkripsi default.

- Amerika Utara: `us-east-1-atl-1aus-east-1-bos-1a`,`us-east-1-chi-1a`,`us-east-1-dfw-1a`, `us-east-1-iah-1aus-east-1-mci-1a`,`us-east-1-mia-1a`,`us-east-1-msp-1a`,`us-east-1-nyc-1a`, `us-east-1-phl-1aus-east-1-qro-1a`,`us-east-2-`

mci-1a,us-east-2-mci-1b,us-west-2-den-1a,us-west-2-hnl-1a, us-west-2-las-1aus-west-2-las-1b,us-west-2-pdx-1a,us-west-2-phx-1a, us-west-2-sea-1a

- Asia Pasifik:ap-northeast-1-tpe-1a,ap-south-1-ccu-1a,ap-south-1-ccu-2a,ap-south-1-del-1a,ap-south-1-del-2a,ap-southeast-1-bkk-1a,ap-southeast-1-han-1a,ap-southeast-1-mnl-1a,ap-southeast-2-akl-1a, ap-southeast-2-per-1a
- Eropa: eu-central-1-ath-1aeu-central-1-ham-1a,eu-central-1-ist-1a,eu-central-1-waw-1a,eu-north-1-cph-1a, eu-north-1-hel-1a
- Amerika Selatan:us-east-1-bue-1a,us-east-1-lim-1a, us-east-1-scl-1a
- Timur Tengah: me-south-1-mct-1a
- Afrika: af-south-1-los-1a

Di semua Zona Lokal lainnya, enkripsi Amazon EBS mengikuti perilaku yang sama seperti di Wilayah induk.

Pos Terdepan dan Zona Panjang Gelombang

Semua sumber daya Amazon EBS di AWS Outposts dan di Zona Panjang Gelombang dienkripsi secara default. Anda tidak dapat membuat volume atau snapshot yang tidak dienkripsi. Jika Anda tidak menentukan kunci KMS, Amazon EBS mengenkripsi sumber daya Anda dengan kunci enkripsi default akun Anda. Secara default, ini adalah Kunci yang dikelola AWS (aws/ebs). Untuk mengubah kunci enkripsi default untuk akun Anda, lihatKunci enkripsi default.

Saat Anda membuat volume dari snapshot lokal di zona ini, volume dienkripsi menggunakan kunci KMS yang sama dengan snapshot sumber. Namun, Anda dapat mengubah kunci enkripsi saat menyalin snapshot lokal.

Persyaratan untuk enkripsi Amazon EBS

Sebelum memulai, verifikasi bahwa persyaratan berikut dipenuhi.

Persyaratan

- Tipe volume yang mendukung
- Tipe instans yang didukung
- Izin untuk pengguna
- Izin untuk instans

Tipe volume yang mendukung

Enkripsi mendukung oleh semua tipe volume EBS. Anda dapat mengharapkan performa IOPS yang sama pada volume terenkripsi seperti pada volume yang tidak terenkripsi, dengan efek minimal pada latensi. Anda dapat mengakses volume terenkripsi dengan cara yang sama seperti Anda mengakses volume yang tidak terenkripsi. Enkripsi dan dekripsi ditangani secara transparan, dan tidak memerlukan tindakan tambahan dari Anda atau aplikasi Anda.

Tipe instans yang didukung

Enkripsi Amazon EBS tersedia di semua jenis instans generasi saat ini dan generasi sebelumnya.

Izin untuk pengguna

Saat Anda menggunakan kunci KMS untuk enkripsi EBS, kebijakan kunci KMS memungkinkan setiap pengguna dengan akses ke AWS KMS tindakan yang diperlukan untuk menggunakan kunci KMS ini untuk mengenkripsi atau mendekripsi sumber daya EBS. Anda harus memberikan izin kepada pengguna untuk melakukan tindakan berikut agar dapat menggunakan enkripsi EBS:

- `kms:CreateGrant`
- `kms:Decrypt`
- `kms:DescribeKey`
- `kms:GenerateDataKeyWithoutPlaintext`
- `kms:ReEncrypt`

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "kms:CreateGrant",
            "Resource": [
                "arn:aws:kms:us-east-2:123456789012:key/abcd1234-a123-456d-a12b-
a123b4cd56ef"
            ],
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": true
                }
            }
        }
    ]
}
```

Untuk informasi selengkapnya, lihat Mengiz inkan akses ke AWS akun dan mengaktifkan kebijakan IAM di bagian Kebijakan kunci default di Panduan Peng AWS Key Management Service embang.

Izin untuk instans

Saat instans mencoba berinteraksi dengan AMI, volume, atau snapshot terenkripsi, pemberian kunci KMS dikeluarkan untuk peran khusus identitas instans. Peran khusus identitas adalah peran IAM yang digunakan oleh instans untuk berinteraksi dengan AMI, volume, atau snapshot yang terenkripsi atas nama Anda.

Identity-only peran tidak perlu dibuat atau dihapus secara manual, dan mereka tidak memiliki kebijakan yang terkait dengannya. Selain itu, Anda tidak dapat mengakses kredensial peran khusus identitas.

> **Note**
>
> Identity-only peran tidak digunakan oleh aplikasi pada instans Anda untuk mengakses sumber daya AWS KMS terenkripsi lainnya, seperti objek Amazon S3 atau tabel Dynamo DB. Operasi ini dilakukan menggunakan kredensional peran instans Amazon EC2, atau AWS kredenSIAL lain yang telah Anda konfigurasi pada instans Anda.

Identity-only peran tunduk pada kebijakan kontrol layanan (SCP), dan kebijakan kunci KMS. Jika kunci SCP atau KMS menolak akses peran identitas saja ke kunci KMS, Anda mungkin gagal meluncurkan instans EC2 dengan volume terenkripsi, atau menggunakan AMI atau snapshot terenkripsi.

Jika Anda membuat SCP atau kebijakan kunci yang menolak akses berdasarkan lokasi jaringan menggunakan kunci kondisi`aws:SourceIp`,`aws:VpcSourceIp`,`aws:SourceVpc`, atau `aws:SourceVpce` AWS global, maka Anda harus memastikan bahwa pernyataan kebijakan ini tidak berlaku untuk peran khusus instance. Untuk contoh kebijakan, lihat Contoh Kebijakan Perimeter Data.

Identity-only peran ARN menggunakan format berikut:

```
arn:aws-partition:iam::account_id:role/aws:ec2-infrastructure/instance_id
```

Ketika pemberian kunci diberikan kepada sebuah instans, pemberian kunci tersebut dikeluarkan untuk sesi peran yang diasumsikan khusus untuk instans tersebut. ARN pengguna utama penerima menggunakan format berikut:

```
arn:aws-partition:sts::account_id:assumed-role/aws:ec2-infrastructure/instance_id
```

Aktifkan enkripsi Amazon EBS secara default

Anda dapat mengonfigurasi AWS akun Anda untuk menerapkan enkripsi volume EBS baru dan salinan snapshot yang Anda buat. Misalnya, Amazon EBS mengenkripsi volume EBS yang dibuat saat Anda meluncurkan instans dan snapshot yang Anda salin dari snapshot yang tidak dienkripsi. Untuk contoh transisi dari sumber daya EBS tidak terenkripsi menjadi terenkripsi, lihat Mengenkripsi sumber daya yang tidak terenkripsi.

Enkripsi secara default tidak berpengaruh pada volume atau snapshot EBS yang ada.

Pertimbangan-pertimbangan

- Enkripsi secara default adalah Region-specific pengaturan. Jika Anda aktifkan untuk sebuah Wilayah, Anda tidak dapat menonaktifkannya untuk volume atau snapshot individual di Wilayah tersebut.
- Enkripsi Amazon EBS secara default didukung pada semua jenis instans generasi saat ini dan generasi sebelumnya.

- Jika Anda menyalin snapshot dan mengenkripsinya ke kunci KMS baru, salinan lengkap (tidak inkremental) dibuat. Hal ini menyebabkan biaya penyimpanan tambahan.

Console

Untuk mengaktifkan enkripsi secara default untuk Wilayah

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi, pilih Wilayah.
3. Dari panel navigasi, pilih Pengaturan, lalu pilih tab Perlindungan dan keamanan data.
4. Di bagian enkripsi EBS, pilih Kel ola.
5. Pilih Aktifkan. Anda menyimpannya Kunci yang dikelola AWS dengan alias yang `aws/ebs` dibuat atas nama Anda sebagai kunci enkripsi default, atau memilih kunci enkripsi yang dikelola pelanggan simetris.
6. Pilih Perbarui enkripsi EBS.

AWS CLI

Untuk melihat pengaturan enkripsi secara default

Gunakan perintah get-ebs-encryption-by-default.

- Untuk Wilayah tertentu

```
aws ec2 get-ebs-encryption-by-default --region region
```

- Untuk semua Wilayah di akun Anda

```
echo -e "Region      \t Encrypt \t Key"; \
echo -e "----------- \t ------- \t -------" ; \
for region in $(aws ec2 describe-regions --region us-east-1 --query "Regions[*].
[RegionName]" --output text);
do
    default=$(aws ec2 get-ebs-encryption-by-default --region $region --query
 "{Encryption_By_Default:EbsEncryptionByDefault}" --output text);
    kms_key=$(aws ec2 get-ebs-default-kms-key-id --region $region | jq
 '.KmsKeyId');
    echo -e "$region \t $default \t\t $kms_key";
done
```

Untuk mengaktifkan enkripsi secara default

Gunakan perintah enable- ebs-encryption-by-default.

- Untuk Wilayah tertentu

```
aws ec2 enable-ebs-encryption-by-default --region region
```

- Untuk semua Wilayah di akun Anda

```
echo -e "Region      \t Encrypt \t Key"; \
echo -e "----------- \t ------- \t -------" ; \
for region in $(aws ec2 describe-regions --region us-east-1 --query "Regions[*].
[RegionName]" --output text);
do
    default=$(aws ec2 enable-ebs-encryption-by-default --region $region --query
 "{Encryption_By_Default:EbsEncryptionByDefault}" --output text);
    kms_key=$(aws ec2 get-ebs-default-kms-key-id --region $region | jq
 '.KmsKeyId');
    echo -e "$region \t $default \t\t $kms_key";
done
```

Untuk menonaktifkan enkripsi secara default

Gunakan perintah disable- ebs-encryption-by-default.

- Untuk Wilayah tertentu

```
aws ec2 disable-ebs-encryption-by-default --region region
```

- Untuk semua Wilayah di akun Anda

```
echo -e "Region      \t Encrypt \t Key"; \
echo -e "----------- \t ------- \t -------" ; \
for region in $(aws ec2 describe-regions --region us-east-1 --query "Regions[*].
[RegionName]" --output text);
do
    default=$(aws ec2 disable-ebs-encryption-by-default --region $region --query
 "{Encryption_By_Default:EbsEncryptionByDefault}" --output text);
    kms_key=$(aws ec2 get-ebs-default-kms-key-id --region $region | jq
 '.KmsKeyId');
    echo -e "$region \t $default \t\t $kms_key";
```

```
done
```

PowerShell

Untuk melihat pengaturan enkripsi secara default

Gunakan Get-EC2EbsEncryptionByDefault cmdlet.

- Untuk Wilayah tertentu

```
Get-EC2EbsEncryptionByDefault -Region region
```

- Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName |
    ForEach-Object {
    [PSCustomObject]@{
        Region                    = $_
        EC2EbsEncryptionByDefault = Get-EC2EbsEncryptionByDefault -Region $_
        EC2EbsDefaultKmsKeyId     = Get-EC2EbsDefaultKmsKeyId -Region $_
    } } |
    Format-Table -AutoSize
```

Untuk mengaktifkan enkripsi secara default

Gunakan Enable-EC2EbsEncryptionByDefault cmdlet.

- Untuk Wilayah tertentu

```
Enable-EC2EbsEncryptionByDefault -Region region
```

- Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName |
    ForEach-Object {
    [PSCustomObject]@{
        Region                    = $_
        EC2EbsEncryptionByDefault = Enable-EC2EbsEncryptionByDefault -Region $_
        EC2EbsDefaultKmsKeyId     = Get-EC2EbsDefaultKmsKeyId -Region $_
    } } |
```

```
    Format-Table -AutoSize
```

Untuk menonaktifkan enkripsi secara default

Gunakan Disable-EC2EbsEncryptionByDefault cmdlet.

- Untuk Wilayah tertentu

```
Disable-EC2EbsEncryptionByDefault -Region region
```

- Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName |
    ForEach-Object {
    [PSCustomObject]@{
        Region                     = $_
        EC2EbsEncryptionByDefault = Disable-EC2EbsEncryptionByDefault -Region $_
        EC2EbsDefaultKmsKeyId     = Get-EC2EbsDefaultKmsKeyId -Region $_
    } } |
    Format-Table -AutoSize
```

Anda tidak dapat mengubah kunci KMS yang terkait dengan snapshot yang ada atau volume terenkripsi. Namun, Anda dapat mengaitkan kunci KMS yang berbeda selama operasi salinan snapshot sehingga snapshot salinan yang dihasilkan dienkripsi oleh kunci KMS yang baru.

Kunci enkripsi default

Amazon EBS secara otomatis membuat unik Kunci yang dikelola AWS di setiap Wilayah tempat Anda membuat sumber daya Amazon EBS. Alias untuk kunci KMS ini adalah`aws/ebs`. Ini adalah kunci enkripsi default Anda. Anda dapat mengubah kunci enkripsi default menjadi kunci yang dikelola pelanggan.

Untuk mengubah kunci enkripsi default untuk Wilayah, gunakan salah satu metode berikut.

Console

Untuk mengubah kunci enkripsi default untuk Wilayah

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi, pilih Wilayah.

3. Dari panel navigasi, pilih Pengaturan, lalu pilih tab Perlindungan dan keamanan data.
4. Di bagian enkripsi EBS, pilih Kel ola.
5. Untuk Kunci enkripsi default, pilih ID kunci KMS yang ingin Anda gunakan.
6. Pilih Perbarui enkripsi EBS.

AWS CLI

Gunakan perintah modify-ebs-default-kms-key-id.

```
aws ec2 modify-ebs-default-kms-key-id --kms-key-id key-id --region region
```

PowerShell

Gunakan Edit-EC2EbsDefaultKmsKeyId cmdlet.

```
Edit-EC2EbsDefaultKmsKeyId -KmsKeyId key-id -Region region
```

Enkripsi sumber daya EBS

Anda mengenkripsi volume EBS dengan mengaktifkan enkripsi, menggunakan enkripsi secara default atau dengan mengaktifkan enkripsi saat Anda membuat volume yang ingin Anda enkripsi.

Saat Anda mengenkripsi volume, Anda dapat menentukan kunci KMS enkripsi simetris untuk mengenkripsi volume. Jika Anda tidak menentukan kunci KMS, kunci KMS yang digunakan untuk enkripsi tergantung pada kondisi enkripsi snapshot sumber dan kepemilikannya. Untuk informasi selengkapnya, lihat tabel hasil enkripsi.

> **Note**
>
> Jika Anda menggunakan API atau AWS CLI untuk menentukan kunci KMS, ketahuilah bahwa itu meng AWS autentikasi kunci KMS secara asinkron. Jika Anda menentukan ID kunci KMS, suatu alias, atau ARN yang tidak valid, tindakan dapat muncul untuk diselesaikan, tetapi akhirnya akan gagal.

Anda tidak dapat mengubah kunci KMS yang terkait dengan snapshot atau volume yang ada. Namun, Anda dapat mengaitkan kunci KMS yang berbeda selama operasi salinan snapshot sehingga snapshot salinan yang dihasilkan dienkripsi oleh kunci KMS yang baru.

Enkripsi volume kosong pada saat pembuatan

Saat Anda membuat volume EBS baru yang kosong, Anda dapat mengenkripsinya dengan mengaktifkan enkripsi untuk operasi pembuatan volume tertentu. Jika Anda mengaktifkan enkripsi EBS secara default, volume akan dienkripsi secara otomatis menggunakan kunci KMS default untuk enkripsi EBS. Sebagai alternatif, Anda dapat menentukan kunci KMS enkripsi simetris yang berbeda untuk operasi pembuatan volume spesifik. Volume dienkripsi saat pertama kali tersedia, sehingga data Anda selalu aman. Untuk prosedur terperinci, lihat Buat volume Amazon EBS.

Secara default, kunci KMS yang Anda pilih saat membuat volume mengenkripsi snapshot yang Anda buat dari volume dan volume yang Anda pulihkan dari snapshot yang dienkripsi tersebut. Anda tidak dapat menghapus enkripsi dari volume atau snapshot terenkripsi, yang berarti bahwa volume yang dipulihkan dari snapshot terenkripsi, atau salinan snapshot terenkripsi, selalu dienkripsi.

Snapshot publik dari volume terenkripsi tidak didukung, tetapi Anda dapat berbagi snapshot terenkripsi dengan akun tertentu. Untuk petunjuk terperinci, lihat Bagikan snapshot Amazon EBS dengan orang lain AWS rekening.

Mengenkripsi sumber daya yang tidak terenkripsi

Anda tidak dapat secara langsung mengenkripsi volume atau snapshot yang tidak terenkripsi yang ada.

Untuk mengenkripsi volume yang tidak dienkripsi, buat snapshot volume itu, lalu gunakan snapshot untuk membuat volume terenkripsi baru. Untuk informasi selengkapnya, lihat Membuat snapshot dan Membuat volume.

Untuk mengenkripsi snapshot yang tidak dienkripsi, buat salinan terenkripsi dari snapshot tersebut. Untuk informasi selengkapnya, lihat Menyalin snapshot.

Jika Anda mengaktifkan akun untuk enkripsi secara default, volume dan salinan snapshot yang dibuat dari snapshot yang tidak terenkripsi selalu dienkripsi. Jika tidak, Anda harus menentukan parameter enkripsi dalam permintaan. Untuk informasi selengkapnya, lihat Aktifkan enkripsi secara default.

Putar AWS KMS kunci yang digunakan untuk enkripsi Amazon EBS

Praktik terbaik kriptografi mencegah penggunaan ulang kunci enkripsi secara ekstensif.

Untuk membuat materi kriptografi baru untuk digunakan dengan enkripsi Amazon EBS, Anda dapat membuat kunci yang dikelola pelanggan baru, lalu mengubah aplikasi Anda untuk menggunakan kunci KMS baru tersebut. Atau, Anda dapat mengaktifkan rotasi kunci otomatis untuk kunci yang dikelola pelanggan yang ada.

Saat Anda mengaktifkan rotasi kunci otomatis untuk kunci yang dikelola pelanggan, AWS KMS menghasilkan materi kriptografi baru untuk kunci KMS setiap tahun. AWS KMS menyimpan semua versi sebelumnya dari materi kriptografi sehingga Anda dapat terus mendekripsi dan menggunakan volume dan snapshot yang sebelumnya dienkripsi dengan materi kunci KMS tersebut. AWS KMS tidak menghapus materi kunci yang diputar sampai Anda menghapus kunci KMS.

Saat Anda menggunakan kunci yang dikelola pelanggan yang dirotasi untuk mengenkripsi volume atau snapshot baru, AWS KMS menggunakan materi kunci (baru) saat ini. Saat Anda menggunakan kunci yang dikelola pelanggan yang dirotasi untuk mendekripsi volume atau snapshot, AWS KMS menggunakan versi materi kriptografi yang digunakan untuk mengenkripsinya. Jika volume atau snapshot dienkripsi dengan versi sebelumnya dari materi kriptografi, AWS KMS terus gunakan versi sebelumnya untuk mendekripsinya. AWS KMS tidak mengenkripsi ulang volume atau snapshot yang dienkripsi sebelumnya untuk menggunakan materi kriptografi baru setelah rotasi kunci. Mereka tetap dienkripsi dengan materi kriptografi yang awalnya dienkripsi. Anda dapat menggunakan kunci yang dikelola pelanggan yang dirotasi dengan aman dalam aplikasi dan AWS layanan tanpa perubahan kode.

> **Note**
>
> - Rotasi kunci otomatis hanya didukung untuk kunci simetris yang dikelola pelanggan dengan bahan kunci yang dibuat AWS KMS .
> - AWS KMS secara otomatis berputar Kunci yang dikelola AWS setiap tahun. Anda tidak dapat mengaktifkan atau menonaktifkan rotasi kunci untuk Kunci yang dikelola AWS.

Untuk informasi selengkapnya, lihat Merotasi kunci KMS di Panduan Developer AWS Key Management Service .

Contoh enkripsi Amazon EBS

Saat Anda membuat sumber daya EBS terenkripsi, sumber daya tersebut dienkripsi dengan kunci KMS default akun Anda untuk enkripsi EBS kecuali Anda menentukan kunci yang dikelola pelanggan yang berbeda dalam parameter pembuatan volume atau pemetaan perangkat blok untuk AMI atau instans.

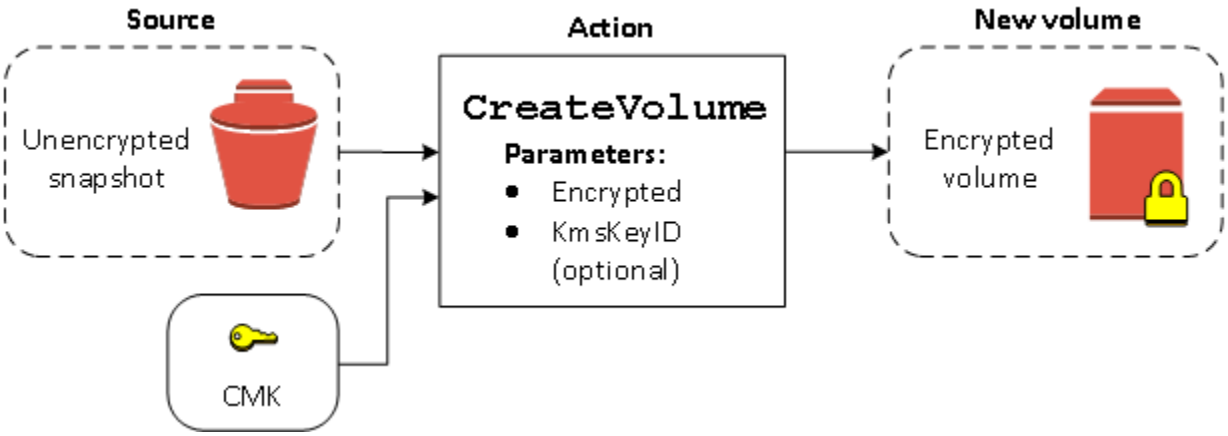
Contoh berikut ini menggambarkan cara mengelola status enkripsi volume dan snapshot Anda. Untuk daftar lengkap kasus enkripsi, lihat tabel hasil enkripsi.

Contoh

- Mengembalikan volume yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)
- Mengembalikan volume yang tidak terenkripsi (enkripsi secara default diaktifkan)
- Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)
- Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)
- Re-encrypt volume terenkripsi
- Re-encrypt snapshot terenkripsi
- Memigrasikan data antara volume terenkripsi dan tidak terenkripsi
- Hasil enkripsi

Mengembalikan volume yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)

Tanpa enkripsi yang diaktifkan secara default, volume yang dipulihkan dari snapshot yang tidak dienkripsi tidak akan dienkripsi secara default. Namun, Anda dapat mengenkripsi volume yang dihasilkan dengan mengatur `Encrypted` dan, secara opsional, `KmsKeyId` parameter. Diagram berikut menggambarkan prosesnya.

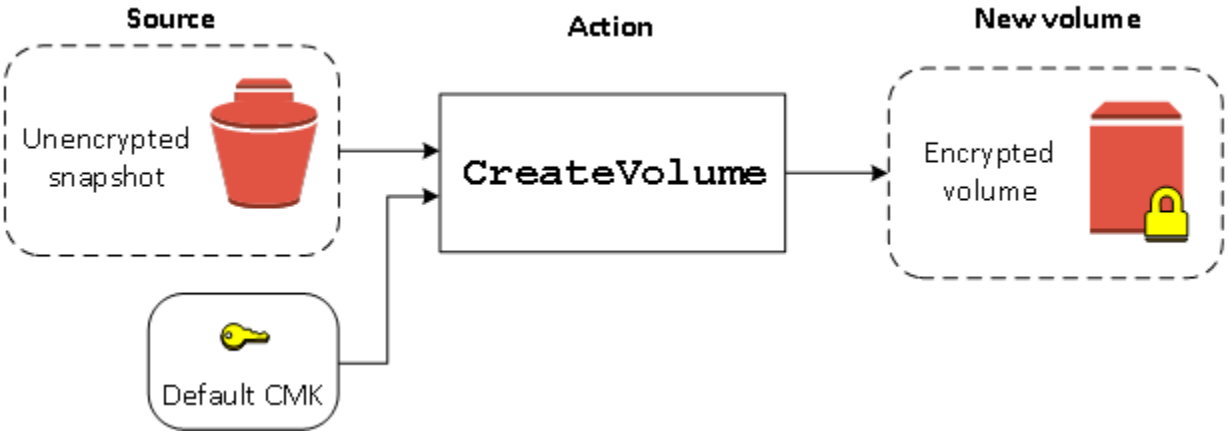


Jika Anda meninggalkan parameter `KmsKeyId`, volume yang dihasilkan dienkripsi menggunakan kunci KMS default Anda untuk enkripsi EBS. Anda harus menentukan ID kunci KMS untuk mengenkripsi volume ke kunci KMS yang berbeda.

Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.

Mengembalikan volume yang tidak terenkripsi (enkripsi secara default diaktifkan)

Jika Anda telah mengaktifkan enkripsi secara default, enkripsi wajib dilakukan untuk volume yang dipulihkan dari snapshot yang tidak terenkripsi, dan tidak ada parameter enkripsi yang diperlukan agar kunci KMS default Anda dapat digunakan. Diagram berikut menunjukkan kasus default sederhana ini:

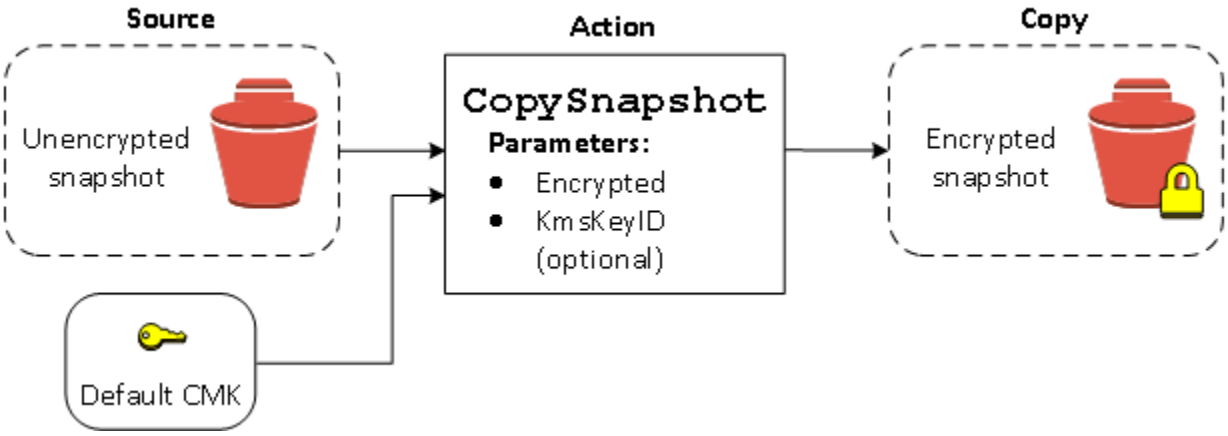


Jika Anda ingin mengenkripsi volume yang dipulihkan ke kunci enkripsi yang dikelola pelanggan simetris, Anda harus menyediakan `Encrypted` dan `KmsKeyId` parameter seperti ditunjukkan dalam Mengembalikan volume yang tidak terenkripsi (enkripsi secara default tidak diaktifkan).

Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)

Tanpa enkripsi yang diaktifkan secara default, salinan snapshot yang tidak dienkripsi tidak akan dienkripsi secara default. Namun, Anda dapat mengenkripsi snapshot yang dihasilkan dengan mengatur parameter `Encrypted` dan, secara opsional, parameter `KmsKeyId`. Jika Anda menghilangkan `KmsKeyId`, snapshot yang dihasilkan dienkripsi oleh kunci KMS default Anda. Anda harus menentukan ID kunci KMS untuk mengenkripsi volume ke kunci KMS enkripsi simetris yang berbeda.

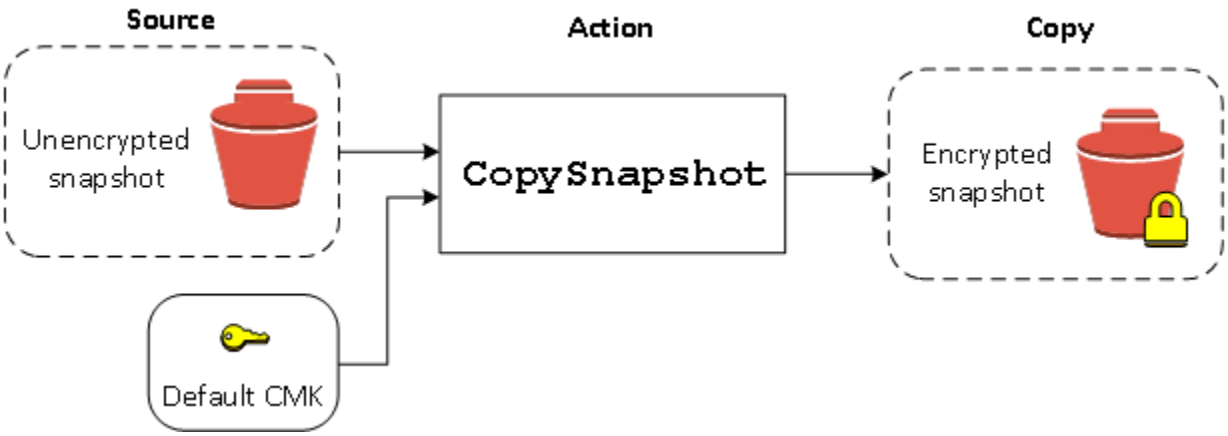
Diagram berikut menggambarkan prosesnya.



Anda dapat mengenkripsi volume EBS dengan menyalin snapshot yang tidak dienkripsi ke snapshot yang dienkripsi, lalu membuat volume dari snapshot yang dienkripsi. Untuk informasi selengkapnya, lihat Menyalin snapshot Amazon EBS.

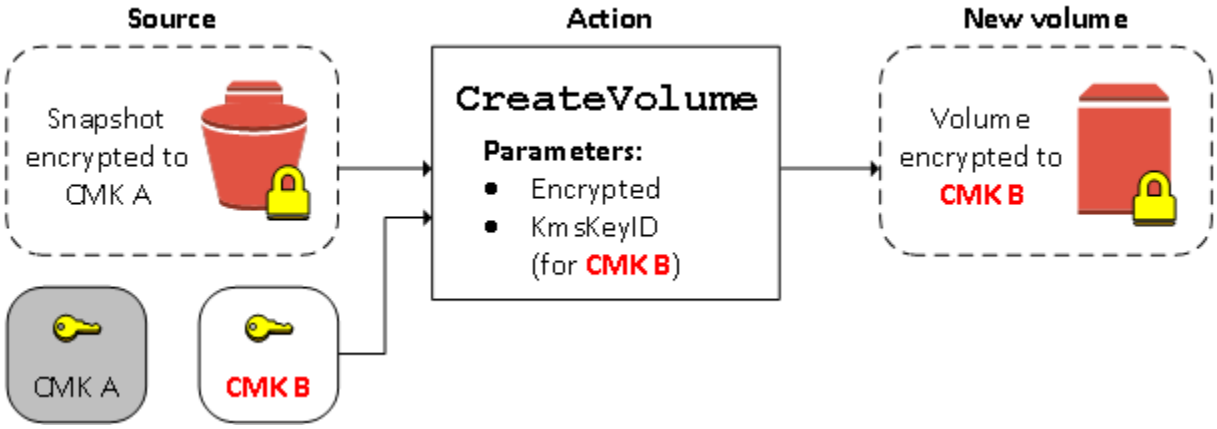
Menyalin snapshot yang tidak terenkripsi (enkripsi secara default tidak diaktifkan)

Ketika Anda telah mengaktifkan enkripsi secara default, enkripsi diwajibkan untuk salinan snapshot yang tidak dienkripsi, dan tidak ada parameter enkripsi yang diperlukan jika kunci KMS default Anda digunakan. Diagram berikut menggambarkan kasus default ini:



Re-encrypt volume terenkripsi

Saat tindakan `CreateVolume` beroperasi pada snapshot terenkripsi, Anda memiliki opsi mengenkripsi ulang kunci KMS yang berbeda. Diagram berikut menggambarkan prosesnya. Dalam contoh ini, Anda memiliki dua kunci KMS, kunci KMS A dan kunci KMS B. Snapshot sumber dienkripsi oleh kunci KMS A. Selama pembuatan volume, dengan ID kunci KMS dari kunci KMS B ditentukan sebagai parameter, data sumber adalah didekripsi secara otomatis, kemudian dienkripsi ulang dengan kunci KMS B.

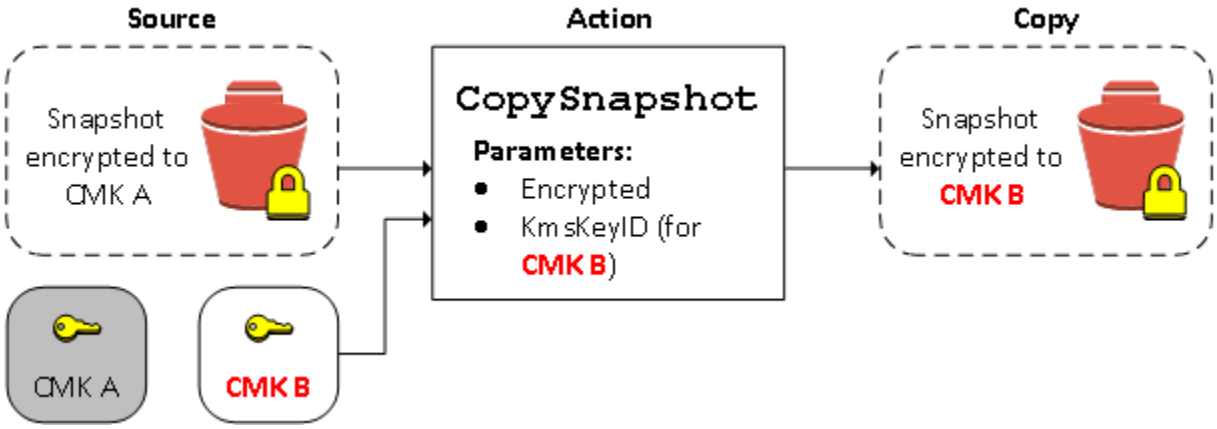


Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.

Re-encrypt snapshot terenkripsi

Kemampuan untuk mengenkripsi snapshot selama penyalinan memungkinkan Anda menerapkan kunci KMS enkripsi simetris baru ke snapshot yang sudah terenkripsi yang Anda miliki. Volume yang dipulihkan dari salinan hasil hanya dapat diakses menggunakan kunci KMS baru. Diagram berikut menggambarkan prosesnya. Dalam contoh ini, Anda memiliki dua kunci KMS, kunci KMS A dan kunci KMS B. Snapshot sumber dienkripsi oleh kunci KMS A. Selama penyalinan, dengan ID kunci

KMS dari kunci KMS B ditentukan sebagai parameter, data sumber secara otomatis dienkripsi ulang dengan kunci KMS B.



Dalam skenario terkait, Anda dapat memilih untuk menerapkan parameter enkripsi baru ke salinan snapshot yang telah dibagikan dengan Anda. Secara default, salinan tersebut dienkripsi dengan kunci KMS yang dibagikan oleh pemilik snapshot. Namun, sebaiknya buat salinan snapshot yang dibagikan menggunakan kunci KMS lain yang Anda kontrol. Hal ini melindungi akses Anda ke volume jika kunci KMS awal terancam, atau jika pemilik mencabut kunci KMS karena alasan apa pun. Untuk informasi selengkapnya, lihat Enkripsi dan penyalinan snapshot.

Memigrasikan data antara volume terenkripsi dan tidak terenkripsi

Saat Anda memiliki akses ke volume terenkripsi dan tidak terenkripsi, Anda dapat dengan bebas mentransfer data di antara keduanya. EC2 menjalankan operasi enkripsi dan dekripsi secara transparan.

Instans Linux

Misalnya, gunakan perintah rsync untuk menyalin data. Dalam perintah berikut, data sumber terletak di `/mnt/source` dan volume tujuan dipasang pada `/mnt/destination`.

```
[ec2-user ~]$ sudo rsync -avh --progress /mnt/source/ /mnt/destination/
```

Instans Windows

Misalnya, gunakan perintah robocopy untuk menyalin data. Dalam perintah berikut, data sumber terletak di `D:\` dan volume tujuan dipasang pada `E:\`.

```
PS C:\> robocopy D:\sourcefolder E:\destinationfolder /e /copyall /eta
```

Kami menyarankan untuk menyalin dari folder, daripada seluruh volume, untuk menghindari potensi masalah dari folder tersembunyi.

Hasil enkripsi

Tabel berikut menjelaskan hasil enkripsi untuk setiap kombinasi pengaturan yang memungkinkan.

Apakah enkripsi diaktifkan?	Apakah enkripsi secara default diaktifkan?	Sumber volume	Default (tidak ada kunci dikelola pelanggan yang ditentukan)	Kustom (kunci dikelola pelanggan ditentukan)
Tidak	Tidak	Volume (kosong) baru	Tidak terenkripsi	N/A
Tidak	Tidak	Snapshot tidak terenkripsi yang Anda miliki	Tidak terenkripsi	
Tidak	Tidak	Snapshot terenkripsi yang Anda miliki	Dienkripsi dengan kunci yang sama	
Tidak	Tidak	Snapshot yang tidak terenkripsi yang dibagikan dengan Anda	Tidak terenkripsi	
Tidak	Tidak	Snapshot terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan*	
Ya	Tidak	Volume baru	Dienkripsi secara default dengan kunci yang dikelola pelanggan	Dienkripsi oleh kunci yang dikelola pelanggan yang ditentukan**
Ya	Tidak	Snapshot tidak terenkripsi yang Anda miliki	Dienkripsi secara default	

Apakah enkripsi diaktifkan?	Apakah enkripsi secara default diaktifkan?	Sumber volume	Default (tidak ada kunci dikelola pelanggan yang ditentukan)	Kustom (kunci dikelola pelanggan ditentukan)
			dengan kunci yang dikelola pelanggan	
Ya	Tidak	Snapshot terenkripsi yang Anda miliki	Dienkripsi dengan kunci yang sama	
Ya	Tidak	Snapshot yang tidak terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Ya	Tidak	Snapshot terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Tidak	Ya	Volume (kosong) baru	Dienkripsi secara default dengan kunci yang dikelola pelanggan	N/A
Tidak	Ya	Snapshot tidak terenkripsi yang Anda miliki	Dienkripsi secara default dengan kunci yang dikelola pelanggan	

Apakah enkripsi diaktifkan?	Apakah enkripsi secara default diaktifkan?	Sumber volume	Default (tidak ada kunci dikelola pelanggan yang ditentukan)	Kustom (kunci dikelola pelanggan ditentukan)
Tidak	Ya	Snapshot terenkripsi yang Anda miliki	Dienkripsi dengan kunci yang sama	
Tidak	Ya	Snapshot yang tidak terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Tidak	Ya	Snapshot terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Ya	Ya	Volume baru	Dienkripsi secara default dengan kunci yang dikelola pelanggan	Dienkripsi oleh kunci yang dikelola pelanggan yang ditentukan
Ya	Ya	Snapshot tidak terenkripsi yang Anda miliki	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Ya	Ya	Snapshot terenkripsi yang Anda miliki	Dienkripsi dengan kunci yang sama	

Apakah enkripsi diaktifkan?	Apakah enkripsi secara default diaktifkan?	Sumber volume	Default (tidak ada kunci dikelola pelanggan yang ditentukan)	Kustom (kunci dikelola pelanggan ditentukan)
Ya	Ya	Snapshot yang tidak terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	
Ya	Ya	Snapshot terenkripsi yang dibagikan dengan Anda	Dienkripsi secara default dengan kunci yang dikelola pelanggan	

* Ini adalah kunci yang dikelola pelanggan default yang digunakan untuk enkripsi EBS untuk AWS akun dan Wilayah. Secara default ini unik Kunci yang dikelola AWS untuk EBS, atau Anda dapat menentukan kunci yang dikelola pelanggan.

** Ini adalah kunci yang dikelola pelanggan yang ditentukan untuk volume saat waktu peluncuran. Kunci yang dikelola pelanggan ini digunakan sebagai pengganti kunci yang dikelola pelanggan default untuk AWS akun dan Wilayah.

Kinerja volume Amazon EBS

Beberapa faktor, termasuk I/O karakteristik dan konfigurasi instans dan volume Anda, dapat memengaruhi kinerja Amazon EBS. Jika Anda mengikuti panduan di halaman detail produk Amazon EBS dan Amazon EC2 kami, Anda biasanya akan mencapai kinerja yang baik. Namun, ada beberapa kasus di mana Anda mungkin perlu melakukan penyetelan untuk mencapai kinerja puncak. Kami merekomendasikan Anda untuk menyesuaikan performa dengan informasi dari beban kerja Anda yang sebenarnya, selain tolok ukur, untuk menentukan konfigurasi optimal Anda. Setelah mempelajari dasar-dasar bekerja dengan volume EBS, ada baiknya melihat I/O kinerja yang Anda butuhkan dan opsi Anda untuk meningkatkan kinerja Amazon EBS untuk memenuhi persyaratan tersebut.

AWS pembaruan kinerja jenis volume EBS mungkin tidak segera berlaku pada volume yang ada. Untuk melihat performa penuh pada volume yang lebih lama, Anda mungkin harus melakukan tindakan `ModifyVolume` terlebih dahulu. Untuk informasi selengkapnya, lihat Memodifikasi volume Amazon EBS menggunakan operasi Volume Elastic.

Daftar Isi

- Kiat performa Amazon EBS
- Optimasi Amazon EBS
- Inisialisasi volume Amazon EBS
- Pembobotan bandwidth instans yang dapat dikonfigurasi
- Karakter I/O istik dan pemantauan Amazon EBS
- Konfigurasi Amazon EBS dan RAID
- Tolok ukur volume Amazon EBS

Kiat performa Amazon EBS

Kiat ini menunjukkan praktik terbaik untuk mendapatkan performa optimal dari volume EBS Anda dalam berbagai skenario pengguna.

Gunakan EBS-optimized instance

Pada instans tanpa dukungan EBS-optimized throughput, lalu lintas jaringan dapat bersaing dengan lalu lintas antara instance Anda dan volume EBS Anda; pada EBS-optimized instance, kedua jenis

lalu lintas tetap terpisah. Beberapa konfigurasi EBS-optimized instans dikenakan biaya tambahan (seperti C3, R3, dan M3), sementara yang lain selalu EBS-optimized tanpa biaya tambahan (seperti M4, C4, C5, dan D2). Untuk informasi selengkapnya, lihat Optimasi Amazon EBS.

Konfigurasikan bandwidth instance

Untuk jenis instans yang didukung, Anda dapat mengonfigurasi bobot bandwidth instans untuk meningkatkan bandwidth Amazon EBS sebesar 25 persen menggunakan bobot `ebs-1` bandwidth. Fitur ini memungkinkan Anda mengoptimalkan alokasi sumber daya jaringan instans antara jaringan EBS dan VPC, yang berpotensi meningkatkan kinerja EBS untuk beban kerja. I/O-intensive Untuk informasi selengkapnya, lihat Pembobotan bandwidth instans yang dapat dikonfigurasi.

Memahami cara menghitung performa

Saat Anda mengukur performa volume EBS, penting untuk memahami unit pengukuran yang terlibat dan cara performa dihitung. Untuk informasi selengkapnya, lihat Karakter I/O istik dan pemantauan Amazon EBS.

Memahami beban kerja Anda

Ada hubungan antara kinerja maksimum volume EBS Anda, ukuran dan jumlah I/O operasi, dan waktu yang diperlukan untuk menyelesaikan setiap tindakan. Masing-masing faktor ini (kinerja I/O, dan latensi) mempengaruhi yang lain, dan aplikasi yang berbeda lebih sensitif terhadap satu faktor atau lainnya. Untuk informasi selengkapnya, lihat Tolok ukur volume Amazon EBS.

Waspadai penalti performa saat menginisialisasi volume dari snapshot

Terdapat peningkatan latensi yang signifikan saat Anda pertama kali mengakses setiap blok data pada volume EBS baru yang dibuat dari snapshot. Anda dapat menghindari lonjakan performa ini menggunakan salah satu opsi berikut:

- Akses setiap blok sebelum memasukkan volume ke dalam produksi. Proses ini disebut menginisialisasi (sebelumnya dikenal sebagai pra-pemanasan). Untuk informasi selengkapnya, lihat Secara manual menginisialisasi volume setelah pembuatan.
- Aktifkan pemulihan snapshot cepat pada snapshot untuk memastikan bahwa volume EBS yang dibuat darinya sepenuhnya diinisialisasi saat pembuatan dan langsung memberikan semua kinerja yang disediakan. Untuk informasi selengkapnya, lihat Pemulihan snapshot cepat Amazon EBS.

Faktor yang dapat menurunkan performa HDD

Saat Anda membuat snapshot dari volume HDD Throughput Dioptimalkan (`st1`) atau Cold HDD (`sc1`), performa dapat menurun sejauh nilai acuan volume saat snapshot sedang berlangsung. Perilaku ini khusus untuk tipe volume ini. Faktor lain yang dapat membatasi kinerja termasuk mendorong lebih banyak throughput daripada yang dapat didukung instans, penalti kinerja yang dihadapi saat menginisialisasi volume yang dibuat dari snapshot, dan jumlah volume kecil acak yang berlebihan I/O pada volume. Untuk informasi selengkapnya tentang penghitungan throughput untuk volume HDD, lihat Tipe volume Amazon EBS.

Kinerja Anda juga dapat terpengaruh jika aplikasi Anda tidak mengirimkan I/O permintaan yang cukup. Ini dapat dipantau dengan melihat panjang dan I/O ukuran antrian volume Anda. Panjang antrian adalah jumlah I/O permintaan tertunda dari aplikasi Anda ke volume Anda. Untuk konsistensi maksimum, HDD-backed volume harus mempertahankan panjang antrian (dibulatkan ke bilangan bulat terdekat) 4 atau lebih saat melakukan 1 MiB berurutan I/O. Untuk informasi selengkapnya tentang memastikan kinerja volume yang konsisten, lihat Karakter I/O istik dan pemantauan Amazon EBS

Tingkatkan pembacaan ke depan untuk beban kerja throughput tinggi dan berat baca di st1 dan sc1 (Hanya instans Linux)

Beberapa beban kerja adalah read-heavy dan mengakses perangkat blok melalui cache halaman sistem operasi (misalnya, dari sistem file). Dalam hal ini, untuk mencapai throughput maksimal, kami sarankan Anda mengonfigurasi pengaturan read-ahead menjadi 1 MiB. Ini adalah pengaturan per perangkat blok yang hanya dapat diterapkan ke volume HDD Anda.

Guna memeriksa nilai read-ahead saat ini untuk perangkat blok Anda, gunakan perintah berikut:

```
$ sudo blockdev --report /dev/<device>
```

Informasi perangkat blok dikembalikan dalam format berikut:

```
RO    RA   SSZ   BSZ   StartSec            Size   Device
rw   256   512  4096       4096      8587820544   /dev/<device>
```

Perangkat yang ditampilkan melaporkan nilai read-ahead sebesar 256 (default). Kalikan angka ini dengan ukuran sektor (512 bita) untuk mendapatkan ukuran buffer read-ahead, yang dalam hal ini adalah 128 KiB. Untuk mengatur nilai buffer ke 1 MiB, gunakan perintah berikut:

```
$ sudo blockdev --setra 2048 /dev/<device>
```

Pastikan bahwa pengaturan read-ahead sekarang menampilkan 2.048 dengan menjalankan kembali perintah pertama.

Gunakan pengaturan ini hanya jika beban kerja Anda terdiri dari beban kerja yang besar dan berurutan I/Os. Jika sebagian besar terdiri dari kecil, acak I/Os, pengaturan ini benar-benar akan menurunkan kinerja Anda. Secara umum, jika beban kerja Anda sebagian besar terdiri dari volume kecil atau acak I/Os, Anda harus mempertimbangkan untuk menggunakan volume SSD Tujuan Umum (gp2dangp3) daripada `sc1` volume `st1` atau.

Gunakan kernel Linux modern (Hanya instans Linux)

Gunakan kernel Linux modern dengan dukungan untuk deskriptor tidak langsung. Setiap kernel 3.8 Linux dan di atasnya memiliki dukungan ini, serta setiap instans EC2 generasi saat ini. Jika I/O ukuran rata-rata Anda mencapai atau mendekati 44 KiB, Anda mungkin menggunakan instance atau kernel tanpa dukungan untuk deskriptor tidak langsung. Untuk informasi tentang memperoleh I/O ukuran rata-rata dari CloudWatch metrik Amazon, lihatKarakter I/O istik dan pemantauan Amazon EBS.

Untuk mencapai throughput maksimal pada volume `st1` atau `sc1`, kami sarankan untuk menerapkan nilai 256 pada parameter `xen_blkfront.max` (untuk versi kernel Linux di bawah 4.6) atau parameter `xen_blkfront.max_indirect_segments` (untuk versi kernel Linux 4.6 dan yang lebih tinggi). Parameter yang sesuai dapat diatur di baris perintah boot OS Anda.

Misalnya, di dalam AMI Amazon Linux dengan kernel sebelumnya, Anda dapat menambahkannya ke akhir baris kernel di konfigurasi GRUB yang ditemukan di `/boot/grub/menu.lst`:

```
kernel /boot/vmlinuz-4.4.5-15.26.amzn1.x86_64 root=LABEL=/ console=ttyS0
 xen_blkfront.max=256
```

Untuk kernel berikutnya, perintah akan serupa dengan yang berikut ini:

```
kernel /boot/vmlinuz-4.9.20-11.31.amzn1.x86_64 root=LABEL=/ console=tty1 console=ttyS0
 xen_blkfront.max_indirect_segments=256
```

Boot ulang instans Anda agar pengaturan ini berfungsi.

Untuk informasi selengkapnya, lihat Meng onfigurasi GRUB untuk AMI paravirtual. Distribusi Linux lainnya, terutama yang tidak menggunakan GRUB boot loader, mungkin memerlukan pendekatan yang berbeda untuk menyesuaikan parameter kernel.

Untuk informasi selengkapnya tentang I/O karakteristik EBS, lihat presentasi Amazon EBS: Designing for Performance re:Invent mengenai topik ini.

Gunakan RAID 0 untuk memaksimalkan pemanfaatan sumber daya instans

Beberapa jenis instans dapat mendorong lebih banyak I/O throughput daripada yang dapat Anda sediakan untuk satu volume EBS. Anda dapat menggabungkan beberapa volume dalam konfigurasi RAID 0 untuk menggunakan bandwidth yang tersedia untuk instans ini. Untuk informasi selengkapnya, lihat Konfigurasi Amazon EBS dan RAID.

Pantau kinerja volume Amazon EBS

Anda dapat memantau dan menganalisis kinerja volume Amazon EBS menggunakan Amazon CloudWatch, pemeriksaan status, dan statistik kinerja terperinci EBS. Untuk informasi selengkapnya, lihat CloudWatch Metrik Amazon untuk Amazon EBS dan Statistik kinerja terperinci Amazon EBS.

Optimasi Amazon EBS

Instans yang dioptimalkan Amazon EBS menggunakan tumpukan konfigurasi yang dioptimalkan dan menyediakan kapasitas khusus tambahan untuk Amazon EBS. I/O Pengoptimalan ini memberikan kinerja terbaik untuk volume EBS Anda dengan meminimalkan perselisihan antara Amazon EBS I/O dan lalu lintas lain dari instans Anda.

Instans yang dioptimalkan EBS memberikan bandwidth khusus untuk Amazon EBS. Saat dilampirkan ke instance yang dioptimalkan EBS:

- `io2`Volume Block Express dirancang untuk memberikan latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O . `io2` Volume Block Express juga memberikan latensi outlier yang lebih baik dibandingkan dengan volume Tujuan Umum, mengurangi frekuensi I/Os melebihi 800 mikrodetik lebih dari 10 kali lipat. `io1`dan `io2` volume dirancang untuk memberikan setidaknya 90% dari kinerja IOPS yang disediakan 99,9% dari waktu dalam tahun tertentu.
- `gp2`dan `gp3` volume dirancang untuk memberikan setidaknya 90% dari kinerja IOPS yang disediakan 99% dari waktu dalam tahun tertentu.
- `st1`dan `sc1` volume memberikan setidaknya 90% dari kinerja throughput yang diharapkan 99% dari waktu dalam tahun tertentu.

Non-compliant periode kira-kira didistribusikan secara seragam, menargetkan 99% dari total throughput yang diharapkan setiap jam. Untuk informasi selengkapnya, lihat Tipe volume Amazon EBS.

I/O Ukuran beban kerja Anda akan memengaruhi latensi rata-rata yang diamati karena latensi meningkat dengan I/O ukuran yang lebih besar. Misalnya, volume `io2` Block Express dirancang untuk memberikan latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O .

Untuk informasi selengkapnya, lihat Instans yang dioptimalkan Amazon EBS di Panduan Pengguna Amazon EC2.

Inisialisasi volume Amazon EBS

Saat Anda membuat volume Amazon EBS, baik dari snapshot EBS atau dari volume EBS lain (salinan volume), blok data harus ditulis ke volume sebelum Anda dapat mengaksesnya. Untuk volume yang dibuat dari snapshot, blok data harus diunduh dari Amazon S3 ke volume baru. Untuk salinan volume, blok data harus disalin dari volume sumber ke salinan volume. Proses ini disebut inisialisasi volume. Selama waktu ini, volume yang diinisialisasi mungkin mengalami peningkatan I/O latensi dan penurunan kinerja. Kinerja volume penuh dicapai hanya setelah semua blok penyimpanan diunduh dan ditulis ke volume.

> **Note**
>
> Volume kosong memberikan kinerja maksimumnya segera setelah pembuatan dan tidak memerlukan inisialisasi.

Tingkat inisialisasi volume default berfluktuasi sepanjang proses inisialisasi, yang dapat membuat waktu penyelesaian tidak dapat diprediksi. Untuk meminimalkan dampak kinerja yang terkait dengan inisialisasi volume, Anda dapat menggunakan opsi berikut:

Topik

- Gunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume
- Gunakan snapshot yang diaktifkan untuk pemulihan snapshot cepat
- Secara manual menginisialisasi volume setelah pembuatan
- Memantau status inisialisasi volume Amazon EBS

> **Note**
>
> Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume dan pemulihan snapshot cepat tidak didukung untuk salinan volume. Untuk informasi selengkapnya, lihat Inisialisasi salinan volume.

Gunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume

Saat membuat volume Amazon EBS dari snapshot, Anda dapat secara opsional menentukan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume (tingkat inisialisasi volume) yang berkisar antara 100 hingga 300. MiB/s Jika Anda menentukan tingkat inisialisasi volume, blok snapshot diunduh dari Amazon S3 dan ditulis ke volume pada tingkat yang ditentukan setelah pembuatan. Ini memungkinkan Anda untuk membuat volume yang sepenuhnya diinisialisasi dan berkinerja penuh dalam jumlah waktu yang dapat diprediksi.

Menggunakan tingkat inisialisasi volume sangat berguna ketika Anda membuat beberapa volume secara bersamaan dan Anda memerlukan semuanya untuk diinisialisasi dalam jumlah waktu yang dapat diprediksi.

> **Note**
>
> Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume didukung dengan semua jenis volume Amazon EBS, dan semua jenis instans Amazon EC2, termasuk instans Mac Amazon EC2.

Anda dapat menentukan tingkat inisialisasi volume:

- Untuk permintaan pembuatan volume individu
- Untuk pemetaan perangkat blok volume EBS dalam permintaan peluncuran contoh
- Untuk pemetaan perangkat blok volume EBS dalam template peluncuran
- Untuk volume EBS yang dibuat oleh tugas penggantian volume root
- Untuk volume EBS di cluster Amazon EKS (dibuat oleh EBS CSI Driver) dan cluster Amazon ECS

Topik

- Cara kerjanya

- Pertimbangan-pertimbangan
- Kuota
- Penagihan
- Tentukan tingkat inisialisasi volume

Cara kerjanya

Saat Anda membuat volume dengan tingkat inisialisasi volume, blok snapshot diunduh dari Amazon S3 ke volume pada tingkat yang Anda tentukan.

Jumlah waktu yang dibutuhkan untuk menginisialisasi volume tergantung pada hal berikut:

- Ukuran data snapshot, bukan ukuran volume yang dibuat.

  > **Tip**
  >
  > Untuk menemukan ukuran data snapshot, periksa `FullSnapshotSizeInBytes` bidang di output perintah describe-snapshots, atau bidang Ukuran snapshot penuh di konsol.

- Tingkat inisialisasi volume yang Anda tentukan

Misalnya, jika Anda membuat volume 20 GiB menggunakan snapshot yang memiliki 10 GiB data, dan Anda menentukan tingkat inisialisasi volume 300 MiB/s, volume akan sepenuhnya diinisialisasi dalam waktu sekitar 34,1 detik (10 GiB/300 = 34,1 detik). MiB/s Demikian pula, jika Anda membuat 10 volume dengan snapshot dan tingkat inisialisasi volume yang sama secara bersamaan, semua 10 volume akan sepenuhnya diinisialisasi dalam 34,1 detik.

Pertimbangan-pertimbangan

- Anda dapat menentukan tingkat inisialisasi volume antara 100 dan 300 MiB/s.
- Saat Anda menentukan tingkat inisialisasi volume, biaya dan waktu penyelesaian didasarkan pada ukuran data snapshot (bukan ukuran volume) dan tarif yang Anda tentukan. Untuk informasi selengkapnya, lihat Penagihan.
- Amazon EBS memberikan tingkat rata-rata yang berada dalam 10 persen dari tingkat inisialisasi volume yang Anda tentukan untuk 99 persen waktu.
- Jika Anda menentukan laju inisialisasi volume dan menggunakan snapshot yang diaktifkan untuk pemulihan snapshot cepat, Amazon EBS akan menggunakan laju yang ditentukan alih-alih

pemulihan snapshot cepat. Sebaliknya, untuk menggunakan pemulihan snapshot cepat, jangan tentukan laju inisialisasi volume.

- Jika Amazon EBS tidak dapat menginisialisasi volume pada tingkat inisialisasi volume yang ditentukan karena kendala kapasitas atau karena Anda telah melampaui kuota, permintaan gagal.
- Anda tidak dapat menentukan tingkat inisialisasi volume untuk volume yang dibuat pada AWS Outposts, atau di Zona Lokal atau Zona Panjang Gelombang.

Kuota

Ada batas 5.000 MiB/s pada tingkat inisialisasi volume kumulatif yang dapat Anda minta di seluruh permintaan pembuatan volume bersamaan. Misalnya, Anda dapat membuat 50 permintaan pembuatan volume bersamaan dengan tingkat 100 MiB/s (50 permintaan simultan * 100 MiB/s tarif), atau 25 permintaan bersamaan dengan tingkat 200 MiB/s (25 permintaan simultan * MiB/s tarif 200). Batas ini berlaku per Wilayah. Jika permintaan melebihi batas ini, itu gagal. Tunggu beberapa permintaan yang sedang berlangsung selesai atau minta kenaikan kuota. Untuk informasi selengkapnya, lihat Kuota untuk Amazon EBS.

Penagihan

Saat membuat volume dengan tingkat inisialisasi volume, Anda akan dikenakan tarif per Gib berdasarkan tingkat inisialisasi yang dipilih. Tarif bervariasi menurut Wilayah. Untuk menentukan tarif untuk tingkat inisialisasi yang dipilih, buka harga Amazon EBS, pilih Wilayah Anda, dan temukan bagian Tarif Tersediakan untuk Inisialisasi Volume.

Anda dikenakan biaya berdasarkan ukuran data snapshot penuh, bukan ukuran volume. Misalnya, jika Anda membuat snapshot dari volume 100 GiB yang berisi 50 GiB data tertulis, ukuran data snapshot penuh adalah 50 GiB. Jika Anda menggunakan snapshot tersebut untuk membuat volume dan menentukan tingkat inisialisasi volume, biaya Anda didasarkan pada 50 GiB.

> **Tip**
>
> Untuk menemukan ukuran data snapshot lengkap, periksa `FullSnapshotSizeInBytes` bidang di output perintah describe-snapshots, atau bidang Ukuran snapshot penuh di konsol.

Rumusnya adalah sebagai berikut:

```
Rate for the selected initialization tier in the Region × full snapshot data size
```

Anda akan ditagih jumlah penuh segera setelah volume memasuki `active` status. Permintaan yang gagal tidak ditagih.

Jika Anda menghapus volume sebelum inisialisasi selesai, Anda masih ditagih untuk tingkat inisialisasi volume yang diminta.

Tentukan tingkat inisialisasi volume

Gunakan salah satu metode berikut untuk membuat volume dari snapshot dengan tingkat inisialisasi volume.

Console

Untuk membuat volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Volume, lalu pilih Buat volume.
3. Untuk Tipe volume, pilih tipe volume yang akan dibuat.
4. Untuk Ukuran, masukkan ukuran yang sama dengan atau lebih besar dari ukuran volume snapshot sumber.
5. Untuk Zona Ketersediaan, pilih Zona Ketersediaan tempat pembuatan volume.
6. Untuk ID Snapshot, pilih snapshot yang akan digunakan.
7. Untuk Tingkat inisialisasi Volume, masukkan tingkat `100` — `300` MiB/s. Menentukan tarif dikenakan biaya tambahan. Untuk menggunakan tingkat inisialisasi default atau pemulihan snapshot cepat, jika diaktifkan untuk snapshot yang dipilih, biarkan bidang ini kosong.

   > **Note**
   >
   > Untuk informasi tentang bagaimana kecepatan dan ukuran data snapshot penuh mempengaruhi waktu inisialisasi, lihatCara kerjanya. Untuk informasi harga, lihatPenagihan.

8. Konfigurasikan pengaturan volume lain yang diperlukan dan tambahkan tag sesuai kebutuhan.
9. Pilih Buat volume.

10. Di grid Volume, lihat status dan kemajuan Inisialisasi volume.

AWS CLI

Untuk membuat volume

Gunakan perintah create-volume. Contoh berikut membuat gp3 volume dari snapshot di Zona Ketersediaan yang ditentukan dengan tingkat inisialisasi volume. 300 MiB/s Ganti contoh Region, Availability Zone, dan snapshot ID dengan nilai sumber daya Anda:

```
aws ec2 create-volume \
    --region us-east-1 \
    --availability-zone us-east-1a \
    --volume-type gp3 \
    --snapshot-id snap-0abcdef1234567890 \
    --volume-initialization-rate 300
```

Jika Anda menghilangkan--size, volume menggunakan ukuran volume snapshot sumber. Jika Anda menentukan--size, nilainya harus sama dengan atau lebih besar dari ukuran volume snapshot sumber.

PowerShell

Untuk membuat volume

Gunakan New-EC2Volume cmdlet. Contoh berikut membuat gp3 volume dari snapshot di Zona Ketersediaan yang ditentukan dengan tingkat inisialisasi volume. 300 MiB/s Perintah menyimpan objek volume yang dikembalikan di$volume. Ganti contoh Region, Availability Zone, dan snapshot ID dengan nilai sumber daya Anda:

```
$volume = New-EC2Volume `
    -Region us-east-1 `
    -AvailabilityZone us-east-1a `
    -VolumeType gp3 `
    -SnapshotId snap-0abcdef1234567890 `
    -VolumeInitializationRate 300
```

Memantau inisialisasi

Informasi inisialisasi volume dapat memakan waktu hingga 5 menit untuk memperbarui. Untuk petunjuk tentang pemantauan inisialisasi, lihatMemantau status inisialisasi volume Amazon EBS.

Gunakan snapshot yang diaktifkan untuk pemulihan snapshot cepat

Jika Anda membuat volume dari snapshot yang diaktifkan untuk pemulihan snapshot cepat, volume sepenuhnya diinisialisasi saat pembuatan dan segera memberikan kinerja penuhnya. Untuk informasi selengkapnya tentang menggunakan pemulihan snapshot cepat, lihatPemulihan snapshot cepat Amazon EBS.

Secara manual menginisialisasi volume setelah pembuatan

Anda dapat menginisialisasi volume Amazon EBS secara manual setelah pembuatan untuk membantu meminimalkan dampak kinerja dari inisialisasi volume.

Anda dapat menggunakan prosedur berikut untuk menginisialisasi volume Amazon EBS secara manual setelah pembuatan.

> **Important**
>
> Saat menginisialisasi volume SSD IOPS yang disediakan yang dibuat dari snapshot, kinerja volume mungkin turun di bawah 50 persen dari level yang diharapkan, yang menyebabkan volume menampilkan status dalam pemeriksaan status `warning` Kinerja. I/O Hal ini wajar, dan Anda dapat mengabaikan status `warning` pada volume SSD IOPS yang Tersedia saat Anda menginisialisasinya. Untuk informasi selengkapnya, lihat Pemeriksaan status volume Amazon EBS.

Instans Linux

Untuk menginisialisasi volume yang dibuat dari snapshot di Linux

1. Lampirkan volume yang baru dipulihkan ke instans Linux Anda.
2. Gunakan perintah lsblk untuk mencantumkan perangkat blok pada instans Anda.

```
$ lsblk
NAME  MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvdf  202:80   0  30G  0 disk
xvda1 202:1    0   8G  0 disk /
```

Di sini Anda dapat melihat volume baru, `/dev/xvdf`, terlampir, tetapi tidak terpasang (karena tidak ada jalur yang tercantum di bawah kolom `MOUNTPOINT`).

3. Gunakan utilitas dd atau fio untuk membaca semua blok pada perangkat. Perintah dd diinstal secara default pada sistem Linux, tetapi fio jauh lebih cepat karena memungkinkan pembacaan multialur.

> **Note**
>
> Langkah ini dapat memakan waktu beberapa menit hingga beberapa jam, bergantung pada bandwidth instans EC2, IOPS yang disediakan untuk volume, dan ukuran volume.

[dd] Parameter `if` (file input) harus diatur ke drive yang ingin Anda inisialisasi. Parameter `of` (file output) parameter harus diatur ke perangkat virtual null Linux, `/dev/null`. Parameter `bs` menetapkan ukuran blok operasi baca; untuk performa yang optimal, harus diatur menjadi 1 MB.

> **Important**
>
> Penggunaan yang salah dd dapat dengan mudah menghancurkan data volume. Pastikan untuk mengikuti perintah contoh di bawah ini dengan tepat. Hanya parameter `if=/dev/xvdf` akan bervariasi tergantung pada nama perangkat yang Anda baca.

```
$ sudo dd if=/dev/xvdf of=/dev/null bs=1M
```

[fio] Jika Anda memiliki fio yang diinstal di sistem Anda, gunakan perintah berikut untuk menginisialisasi volume Anda. Parameter `--filename` (file input) harus diatur ke drive yang ingin Anda inisialisasi.

```
$ sudo fio --filename=/dev/xvdf --rw=read --bs=1M --iodepth=32 --ioengine=libaio --direct=1 --name=volume-initialize
```

Untuk menginstal fio di Amazon Linux, gunakan perintah berikut:

```
sudo yum install -y fio
```

Untuk menginstal fio di Ubuntu, gunakan perintah berikut:

```
sudo apt-get install -y fio
```

Setelah operasi selesai, Anda akan melihat laporan operasi yang sudah dibaca. Volume Anda sekarang siap digunakan. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

Instans Windows

Sebelum menggunakan alat, kumpulkan informasi lebih jelas tentang disk pada sistem Anda sebagai berikut:

Untuk mengumpulkan informasi tentang disk sistem

1. Gunakan perintah wmic untuk mencantumkan disk yang tersedia di sistem Anda:

   ```
   wmic diskdrive get size,deviceid
   ```

   Berikut ini adalah output contoh:

   ```
   DeviceID            Size
   \\.\PHYSICALDRIVE2  80517265920
   \\.\PHYSICALDRIVE1  80517265920
   \\.\PHYSICALDRIVE0  128849011200
   \\.\PHYSICALDRIVE3  107372805120
   ```

2. Identifikasi disk untuk menginisialisasi menggunakan dd atau fio. Drive `C:` berada di `\\.\PHYSICALDRIVE0`. Anda dapat menggunakan utilitas `diskmgmt.msc` untuk membandingkan huruf drive dengan nomor drive disk jika Anda tidak yakin nomor drive mana yang harus digunakan.

Use the dd utility

Selesaikan prosedur berikut untuk memasang dan menggunakan dd untuk menginisialisasi volume.

Pertimbangan penting

- Langkah ini dapat memakan waktu beberapa menit hingga beberapa jam, bergantung pada bandwidth instans EC2, IOPS yang disediakan untuk volume, dan ukuran volume.
- Penggunaan dd yang salah dapat dengan mudah menghancurkan data volume. Pastikan untuk mengikuti prosedur ini secara tepat.

Untuk menginstal dd untuk Windows

dd untuk program Windows memberikan pengalaman yang serupa dengan program dd yang umumnya tersedia untuk sistem Linux dan Unix, dan memungkinkan menginisialisasi volume Amazon EBS yang dibuat dari snapshot. Versi beta yang paling terbaru mendukung perangkat virtual `/dev/null`. Jika Anda menginstal versi sebelumnya, Anda dapat menggunakan perangkat virtual `nul` sebagai gantinya. Dokumentasi lengkap tersedia di http://www.chrysocome.net/dd.

1. Unduh versi biner terbaru dd untuk Windows dari http://www.chrysocome.net/dd.
2. (Opsional) Buat folder untuk utilitas baris perintah yang mudah ditemukan dan diingat, seperti `C:\bin`. Jika Anda sudah memiliki folder khusus untuk baris perintah, Anda dapat menggunakan folder tersebut pada langkah berikut.
3. Buka paket biner dan salin file `dd.exe` ke folder utilitas baris perintah (misalnya, `C:\bin`).
4. Tambahkan baris perintah folder ke variabel lingkungan Jalur Anda sehingga Anda dapat menjalankan program di folder tersebut dari mana saja.
   a. Pilih Mulai, buka menu konteks (klik kanan) untuk Komputer, lalu pilih Properti.
   b. Pilih Pengaturan sistem lanjutan, Variabel Lingkungan.
   c. Untuk Variabel Sistem, pilih variabel Jalur dan pilih Edit.
   d. Untuk Nilai variabel, tambahkan titik koma dan lokasi folder utilitas baris perintah (**`;C:\bin\`**) ke akhir nilai yang ada.
   e. Pilih OK untuk menutup jendela Edit Variabel Sistem.
5. Buka jendela prompt perintah baru. Langkah sebelumnya tidak memperbarui variabel lingkungan di jendela prompt perintah Anda saat ini. Jendela perintah yang Anda buka sekarang setelah menyelesaikan langkah sebelumnya diperbarui.

Untuk menginisialisasi suatu volume menggunakan dd untuk Windows

Jalankan perintah berikut untuk membaca semua blok pada perangkat yang ditentukan (dan mengirim output ke perangkat virtual `/dev/null`). Perintah ini menginisialisasi data yang ada secara aman.

```
dd if=\\.\PHYSICALDRIVEn of=/dev/null bs=1M --progress --size
```

Anda mungkin mendapatkan kesalahan jika dd mencoba membaca di luar akhir volume. Anda dapat mengabaikan kesalahan ini dengan aman.

Jika Anda menggunakan versi sebelumnya dari perintah dd,perintah tidak mendukung perangkat `/dev/null`. Sebaliknya, Anda dapat menggunakan pernagkat `nul` seperti berikut.

```
dd if=\\.\PHYSICALDRIVEn of=nul bs=1M --progress --size
```

Use the fio utility

Selesaikan prosedur berikut untuk memasang dan menggunakan fio untuk menginisialisasi volume.

Untuk menginstal fio untuk Windows

fio untuk program Windows memberikan pengalaman yang serupa dengan program fio yang umumnya tersedia untuk sistem Linux dan Unix, dan memungkinkan Anda untuk menginisialisasi volume Amazon EBS yang dibuat dari snapshot. Untuk informasi selengkapnya, lihat https://github.com/axboe/fio.

1. Unduh penginstal MSI fio dengan memperluas Aset untuk rilis terbaru dan memilih penginstal MSI.
2. Instal fio.

Untuk menginisialisasi volume menggunakan fio untuk Windows

1. Jalankan perintah yang mirip dengan berikut ini untuk menginisialisasi volume:

   ```
   fio --filename=\\.\PHYSICALDRIVEn  --rw=read --bs=1M --iodepth=32 --direct=1 --
   name=volume-initialize
   ```

2. Setelah operasi selesai, Anda siap untuk menggunakan volume baru Anda. Untuk informasi selengkapnya, lihat Menyediakan volume Amazon EBS untuk digunakan.

Memantau status inisialisasi volume Amazon EBS

Saat Anda membuat volume, baik dari snapshot atau dari volume lain (salinan volume), Anda dapat memantau status inisialisasi volume untuk menentukan apakah proses inisialisasi selesai. Anda dapat memantau inisialisasi volume menggunakan opsi berikut:

Topik

- AWS CLI, AWS Alat untuk PowerShell, dan konsol Amazon EC2

- Amazon EventBridge

AWS CLI, AWS Alat untuk PowerShell, dan konsol Amazon EC2

Anda dapat menggunakan konsol AWS CLI, AWS Tools forPowerShell, dan Amazon EC2 untuk memeriksa status inisialisasi volume kapan saja setelah volume dibuat. Informasi berikut disediakan:

- Jenis inisialisasi (AWS CLI dan AWS Alat untuk PowerShell saja) — Menunjukkan jenis inisialisasi volume yang digunakan. `default`untuk pemulihan snapshot cepat dan inisialisasi volume default, `provisioned-rate` untuk Amazon EBS Provisioned Rate untuk Inisialisasi Volume, dan `volume-copy` untuk inisialisasi salinan volume.
- Perkiraan waktu penyelesaian (AWS CLI dan AWS Alat PowerShell hanya untuk) — Hanya untuk volume yang dibuat menggunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume. Perkiraan waktu yang tersisa, dalam detik, untuk menyelesaikan inisialisasi volume.
- Kemajuan — Kemajuan, sebagai persentase (0-100), untuk proses inisialisasi volume. Untuk volume yang diinisialisasi dengan pemulihan snapshot cepat, kemajuan bergerak ke 100 persen segera setelah pembuatan.
- Status inisialisasi — Status keseluruhan inisialisasi volume (`initializing`ataucompleted). Untuk volume yang diinisialisasi dengan pemulihan snapshot cepat, status berpindah ke `completed` segera setelah pembuatan.

> **Note**
>
> Diperlukan waktu hingga 5 menit untuk memperbarui informasi inisialisasi volume.

Gunakan tab berikut untuk memeriksa status inisialisasi volume untuk volume Anda.

Console

Untuk memantau status inisialisasi volume

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.
3. Pilih volume yang akan diperiksa status inisialisasi volume.

4. Bid ang Status inisialisasi di tab grid dan Detail memberikan informasi kemajuan dalam format berikut: Status inisialisasi (persentase kemajuan). Misalnya, Inisialisasi (75%).

   Status inisialisasi yang mungkin meliputi: Inisialisasi dan Selesai.

AWS CLI

Untuk memantau status inisialisasi volume

Gunakan AWS CLI perintah describe-volume-status untuk melihat status inisialisasi. Output `EstimatedTimeToCompleteInSeconds` hanya mencakup volume yang dibuat dengan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume.

Misalnya, perintah berikut memeriksa status inisialisasi untuk volume`vol-1111111111111111`, yang dibuat dengan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume.

```
aws ec2 describe-volume-status --volume-ids vol-1111111111111111
```

Berikut ini adalah output contoh.

```
{
    "VolumeStatuses": [
        {
            "Actions": [],
            "AvailabilityZone": "us-east-1a",
            "Events": [],
            "VolumeId": "vol-1111111111111111",
            "VolumeStatus": {
                "Details": [
                    {
                        "Name": "io-enabled",
                        "Status": "passed"
                    },
                    {
                        "Name": "io-performance",
                        "Status": "not-applicable"
                    },
                    {
                        "Name": "initialization-state",
                        "Status": "completed"
                    }
```

```
                ],
                "Status": "ok"
            },
            "InitializationStatusDetails": {
                "InitializationType": "provisioned-rate",
                "Progress": 75,
                "EstimatedTimeToCompleteInSeconds": 850
            }
        }
    ]
}
```

PowerShell

Untuk memantau status inisialisasi volume

Gunakan Get-EC2VolumeStatus cmdlet untuk melihat status inisialisasi. Output `EstimatedTimeToCompleteInSeconds` hanya mencakup volume yang dibuat dengan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume.

Misalnya, perintah berikut memeriksa status inisialisasi untuk volume`vol-1111111111111111`, yang dibuat dengan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume.

```
(Get-EC2VolumeStatus `
    -Region us-east-1 `
    -VolumeId vol-1111111111111111).InitializationStatusDetails |
    Select-Object `
        InitializationType,
        Progress,
        EstimatedTimeToCompleteInSeconds
```

Amazon EventBridge

A EventBridge cara Amazon dikirim ke akun Anda dalam waktu lima menit setelah inisialisasi volume selesai. Anda dapat membuat aturan yang memicu tindakan terprogram sebagai respons terhadap peristiwa ini.

> **Note**
>
> - Peristiwa dipancarkan atas dasar upaya terbaik.

- Jika Anda menghapus volume sebelum inisialisasi selesai, atau dalam waktu 5 menit setelah inisialisasi selesai, Anda mungkin tidak menerima acara tersebut.

Untuk informasi lebih lanjut tentang acara tersebut, lihatAcara inisialisasi volume EBS.

Untuk memantau status inisialisasi volume menggunakan EventBridge

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Pilih Aturan, Buat aturan.
3. Untuk Langkah 1, lakukan hal berikut:
   a. Tentukan nama dan deskripsi untuk aturan.
   b. Untuk bus acara, pilih bus untuk menerima acara. Jika Anda belum membuat bus acara khusus, pertahankan default, atau lihat Membuat bus acara.
   c. Untuk tipe A turan, per tahankan Aturan dengan pola peristiwa.
   d. Pilih Berikutnya.
4. Untuk Langkah 2, lakukan hal berikut:
   a. Untuk sumber acara, simpan AWS acara atau acara EventBridge mitra.
   b. Untuk Metode Pembuatan, pilih Pola kustom (editor JSON).
   c. Untuk P ola A cara, tambahkan yang berikut ini:

```
{
    "detail-type": ["EBS Volume Notification"],
    "source": ["aws.ec2"],
    "detail": {
        "event": ["initializeVolume"],
        "result": ["succeeded"]
    }
}
```

   Untuk contoh acara, lihatAcara inisialisasi volume EBS.
   d. Pilih Berikutnya.
5. Untuk Langkah 3, lakukan hal berikut:
   a. Untuk Jenis target, pilih Layanan AWS .

b. Untuk Pilih target, pilih topik SNS, dan untuk Top ik pilih topik yang diperlukan. Jika Anda belum membuat topik apa pun, lihat Membuat topik.

c. Untuk Iz in, tetap pilih Gunakan peran eksekusi (disarankan).

d. Untuk peran Eksek usi, tetap pilih Buat peran baru untuk sumber daya spesifik ini dan nama peran defauly.

e. Pilih Berikutnya.

6. Untuk Langkah 4, tentukan tag untuk aturan jika diperlukan, lalu pilih Ber ikutnya.
7. Untuk Langkah 5, tinjau aturan lalu pilih Buat aturan.

Pembobotan bandwidth instans yang dapat dikonfigurasi

Konfigurasi bandwidth instans (IBC) adalah fitur yang memungkinkan Anda menyesuaikan alokasi bandwidth jaringan antara jaringan Amazon EBS dan VPC untuk instans Amazon EC2. Fitur ini dapat membantu Anda mengoptimalkan kinerja untuk beban kerja dengan persyaratan bandwidth tertentu. Konfigurasi bandwidth instans hanya didukung pada beberapa instans. Untuk informasi selengkapnya, lihat Konfigur asi bobot bandwidth instans.

Untuk kinerja EBS, menggunakan pembobotan `ebs-1` bandwidth meningkatkan bandwidth EBS dasar sebesar 25 persen sambil mengurangi bandwidth jaringan VPC dengan jumlah absolut yang sama. Ini dapat bermanfaat untuk I/O-intensive beban kerja yang membutuhkan throughput EBS yang lebih tinggi.

Saat merencanakan beban kerja Anda, pertimbangkan I/O ukuran dan pola Anda dengan cermat. I/O Ukuran yang lebih kecil umumnya kurang terpengaruh oleh batasan bandwidth, sementara I/O ukuran yang lebih besar atau beban kerja berurutan dapat mengalami dampak yang lebih signifikan dari perubahan bandwidth. Sangat penting untuk menguji beban kerja spesifik Anda secara menyeluruh untuk memastikan kinerja optimal dengan pembobotan bandwidth yang Anda pilih.

Pertimbangan-pertimbangan

- Bandwidth instans yang dapat dikonfigurasi didukung pada jenis instans tertentu. Untuk informasi selengkapnya, lihat Jenis instans yang didukung.
- Menggunakan pembobotan `ebs-1` bandwidth meningkatkan bandwidth EBS hingga 25 persen, yang dapat meningkatkan kinerja aplikasi. I/O-intensive Namun, perlu diingat bahwa bandwidth jaringan VPC akan berkurang dengan jumlah absolut yang sama (spesifikasi bandwidth gabungan antara EBS dan jaringan tidak berubah).

- Perubahan bobot bandwidth dapat secara signifikan mempengaruhi I/O kinerja. Dengan pembobotan `vpc-1` bandwidth, bandwidth jaringan meningkat, tetapi Anda mungkin mengalami IOPS yang lebih rendah dari yang diharapkan untuk volume EBS. Ini karena Anda mungkin mencapai batas bandwidth EBS sebelum batas IOPS, terutama dengan ukuran yang lebih besar I/O . Misalnya, jenis instans yang biasanya mendukung 240.000 IOPS dengan I/O ukuran 16 KiB mungkin mencapai lebih sedikit IOPS saat menggunakan bobot `vpc-1` bandwidth karena penurunan bandwidth EBS.
- Selalu uji beban kerja spesifik Anda untuk memastikan bahwa pembobotan bandwidth yang Anda pilih memenuhi kebutuhan kinerja Anda.
- Anda dapat mengonfigurasi bobot bandwidth selama peluncuran instance atau memodifikasinya untuk instans yang dihentikan. Untuk informasi selengkapnya, lihat Mengonfigurasi bobot bandwidth untuk instans Anda.
- Anda dapat mengonfigurasi pembobotan bandwidth instans tanpa biaya tambahan.

Karakter I/O istik dan pemantauan Amazon EBS

Pada konfigurasi volume tertentu, I/O karakteristik tertentu mendorong perilaku kinerja untuk volume EBS Anda.

- SSD-backed volume, SSD Tujuan Umum (`gp2`dan`gp3`) dan SSD IOPS Tersediakan (`io1`dan`io2`), memberikan kinerja yang konsisten baik I/O operasi acak atau berurutan.
- HDD-backed volume, Throughput Optimized HDD (`st1`) dan Cold HDD (`sc1`), memberikan kinerja optimal hanya ketika I/O operasi besar dan berurutan.

Untuk memahami bagaimana kinerja volume SSD dan HDD dalam aplikasi Anda, penting untuk mengetahui hubungan antara permintaan pada volume, jumlah IOPS yang tersedia untuknya, waktu yang dibutuhkan untuk menyelesaikan I/O operasi, dan batas throughput volume.

Topik

- IOPS
- Panjang antrean volume dan latensi
- I/O batas throughput ukuran dan volume
- Memantau I/O karakteristik menggunakan CloudWatch
- Memantau statistik I/O kinerja waktu nyata

- Sumber daya terkait

IOPS

IOPS adalah satuan ukuran yang mewakili input/output operasi per detik. Operasi diukur dalam KiB, dan teknologi drive yang mendasarinya menentukan jumlah maksimum data yang dihitung jenis volume sebagai tunggal I/O. I/O Ukuran dibatasi pada 256 KiB untuk volume SSD dan 1.024 KiB untuk volume HDD karena volume SSD menangani volume kecil atau acak I/O jauh lebih efisien daripada volume HDD.

Ketika I/O operasi kecil secara fisik berurutan, Amazon EBS mencoba menggabungkannya menjadi satu I/O operasi hingga I/O ukuran maksimum. Demikian pula, ketika I/O operasi lebih besar dari I/O ukuran maksimum, Amazon EBS mencoba membaginya menjadi I/O operasi yang lebih kecil. Tabel berikut menunjukkan beberapa contoh.

Tipe volume	I/O Ukuran maksimum	I/O operasi dari aplikasi Anda	Jumlah IOPS	Catatan
SSD	256 KiB	1 x 1024 KiB I/O operasi	4 (1.024÷256=4)	Amazon EBS membagi operasi 1.024 KiB menjadi empat I/O operasi 256 KiB yang lebih kecil.
		8 x I/O operasi 32 KiB berurutan	1 (8x32=256)	Amazon EBS menggabungkan delapan operasi 32 KiB berurutan menjadi satu I/O operasi 256 KiB.
		8 I/O operasi 32 KiB acak	8	Amazon EBS menghitung I/O operasi acak secara terpisah.

Tipe volume	I/O Ukuran maksimum	I/O operasi dari aplikasi Anda	Jumlah IOPS	Catatan
HDD	1.024 KiB	1 x 1024 KiB I/O operasi	1	I/O Operasi sudah sama dengan I/O ukuran maksimum. Hal ini tidak digabung atau dibagi.
		8 x I/O operasi 128 KiB berurutan	1 (8x128=1.024)	Amazon EBS menggabungkan delapan operasi 128 KiB berurutan menjadi satu I/O operasi 1.024 KiB. I/O
		8 I/O operasi 32 KiB acak	8	Amazon EBS menghitung I/O operasi acak secara terpisah.

Akibatnya, ketika Anda membuat SSD-backed volume yang mendukung 3.000 IOPS (baik dengan menyediakan `io2` volume `io1` atau dengan 3.000 IOPS, dengan mengukur `gp2` volume pada 1.000 GiB, atau dengan menggunakan `gp3` volume), dan Anda melampirkannya ke EBS-optimized instance yang dapat menyediakan bandwidth yang cukup, Anda dapat mentransfer hingga 3.000 I/Os data per detik, dengan throughput ditentukan berdasarkan ukuran. I/O

Panjang antrean volume dan latensi

Panjang antrian volume adalah jumlah I/O permintaan yang tertunda untuk perangkat. Latensi adalah waktu klien ujung ke ujung yang sebenarnya dari suatu I/O operasi, dengan kata lain, waktu yang berlalu antara mengirim I/O ke EBS dan menerima pengakuan dari EBS bahwa I/O baca atau tulis selesai. Panjang antrian harus dikalibrasi dengan benar dengan I/O ukuran dan latensi untuk menghindari kemacetan baik pada sistem operasi tamu atau pada tautan jaringan ke EBS.

Lama antrean yang optimal bervariasi untuk setiap beban kerja, tergantung pada sensitivitas aplikasi tertentu Anda terhadap IOPS dan latensi. Jika beban kerja Anda tidak mengirimkan I/O permintaan yang cukup untuk sepenuhnya menggunakan kinerja yang tersedia untuk volume EBS Anda, volume Anda mungkin tidak mengirimkan IOPS atau throughput yang telah Anda sediakan.

Transaction-intensive aplikasi sensitif terhadap peningkatan I/O latensi dan sangat cocok untuk SSD-backed volume. Anda dapat mempertahankan IOPS yang tinggi sekaligus menjaga latensi tetap rendah dengan mempertahankan panjang antrean yang rendah dan sejumlah besar IOPS yang tersedia untuk volume. Secara konsisten mendorong lebih banyak IOPS ke volume daripada yang tersedia dapat menyebabkan peningkatan I/O latensi. Untuk konsistensi maksimum, volume harus mempertahankan kedalaman antrian rata-rata (dibulatkan ke bilangan bulat terdekat) sebesar satu untuk setiap 1.000 IOPS yang disediakan dalam satu menit. Misalnya, untuk volume yang disediakan dengan 3.000 IOPS, rata-rata kedalaman antrian harus 3.

Throughput-intensive aplikasi kurang sensitif terhadap peningkatan I/O latensi, dan sangat cocok untuk HDD-backed volume. Anda dapat mempertahankan throughput tinggi ke HDD-backed volume dengan mempertahankan panjang antrian yang tinggi saat melakukan yang besar dan berurutan I/O.

I/O batas throughput ukuran dan volume

Untuk SSD-backed volume, jika I/O ukuran Anda sangat besar, Anda mungkin mengalami jumlah IOPS yang lebih kecil daripada yang Anda sediakan karena Anda mencapai batas throughput volume. Misalnya, gp2 volume di bawah 1.000 GiB dengan kredit burst yang tersedia memiliki batas IOPS 3.000 dan batas throughput volume 250. MiB/s Jika Anda menggunakan I/O ukuran 256 KiB, volume Anda mencapai batas throughput pada 1000 IOPS (1000 x 256 KiB = 250 MiB). Untuk I/O ukuran yang lebih kecil (seperti 16 KiB), volume yang sama ini dapat mempertahankan 3.000 IOPS karena throughput jauh di bawah 250 MiB/s. (Contoh-contoh ini meng I/O asumsikan bahwa volume Anda tidak mencapai batas throughput instans.) Untuk informasi selengkapnya tentang batas throughput untuk setiap tipe volume EBS, lihat Tipe volume Amazon EBS.

Untuk I/O operasi yang lebih kecil, Anda mungkin melihat nilai IOPS yang lebih tinggi dari yang disediakan seperti yang diukur dari dalam instans Anda. Ini terjadi ketika sistem operasi instance menggabungkan operasi kecil menjadi I/O operasi yang lebih besar sebelum meneruskannya ke Amazon EBS.

Jika beban kerja Anda menggunakan `sc1` volume I/Os on HDD-backed `st1` dan sekuensial, Anda mungkin mengalami jumlah IOPS yang lebih tinggi dari yang diharapkan seperti yang diukur dari dalam instans Anda. Ini terjadi ketika sistem operasi instance menggabungkan secara berurutan I/Os dan menghitungnya dalam 1.024 unit KiB-sized . Jika beban kerja Anda menggunakan kecil atau

acak I/Os, Anda mungkin mengalami throughput yang lebih rendah dari yang Anda harapkan. Ini karena setiap penghitungan acak dan non- I/O sekuensial terhadap jumlah IOPS total, yang dapat menyebabkan Anda mencapai batas IOPS volume lebih cepat dari yang diharapkan.

Apa pun jenis volume EBS Anda, jika Anda tidak mengalami IOPS atau throughput yang Anda harapkan dalam konfigurasi Anda, pastikan bandwidth instans EC2 Anda bukan faktor pembatas. Anda harus selalu menggunakan EBS-optimized instance generasi saat ini (atau yang mencakup 10 konektivitas Gb/s jaringan) untuk kinerja optimal. Kemungkinan penyebab lain untuk tidak mengalami IOPS yang diharapkan adalah Anda tidak cukup mengemudi I/O ke volume EBS.

Memantau I/O karakteristik menggunakan CloudWatch

Anda dapat memantau I/O karakteristik ini dengan metrik volume masing-masing CloudWatch volume.

Monitor untuk terhenti I/O

`VolumeStalledIOCheck` memantau status volume EBS Anda untuk menentukan kapan volume Anda terganggu. Metrik adalah nilai biner yang akan mengembalikan status `0` (lulus) atau `1` (gagal) berdasarkan apakah volume EBS dapat menyelesaikan I/O operasi atau tidak.

Jika `VolumeStalledIOCheck` metrik gagal, Anda dapat menunggu AWS untuk menyelesaikan masalah, atau Anda dapat mengambil tindakan, seperti mengganti volume yang terpengaruh atau menghentikan dan memulai ulang instance yang dilampirkan volume. Dalam kebanyakan kasus, ketika metrik ini gagal, EBS akan secara otomatis mendiagnosis dan memulihkan volume Anda dalam beberapa menit. Anda dapat menggunakan I/O tindakan Jeda AWS Fault Injection Service untuk menjalankan eksperimen terkontrol untuk menguji arsitektur dan pemantauan berdasarkan metrik ini untuk meningkatkan ketahanan terhadap kesalahan penyimpanan.

Memantau I/O latensi untuk volume

Anda dapat memantau latensi rata-rata untuk operasi baca dan tulis untuk volume Amazon EBS menggunakan `VolumeAvgWriteLatency` metrik `VolumeAvgReadLatency` dan masing-masing. Anda dapat menggunakan tindakan Injeksi Latensi AWS Fault Injection Service untuk menjalankan eksperimen terkontrol guna menguji arsitektur dan pemantauan berdasarkan metrik ini untuk meningkatkan ketahanan terhadap penurunan kinerja penyimpanan.

Jika I/O latensi Anda lebih tinggi dari yang Anda butuhkan, pastikan aplikasi Anda tidak mencoba mendorong lebih banyak IOPS atau throughput daripada yang telah Anda sediakan untuk volume Anda. Anda dapat menggunakan `VolumeAvgThroughput` metrik `VolumeAvgIOPS` dan untuk

memantau rata-rata IOPS dan throughput yang didorong ke volume Anda dalam satu menit dan kemudian membandingkannya dengan IOPS dan throughput yang disediakan volume. Jika volume tidak mendorong operasi apa pun selama menit, metrik akan melaporkan nilai nol (`0`). Jika semburan IOPS atau throughput tinggi terjadi untuk waktu yang lebih singkat dari interval menit maka volume mengalami ledakan mikro, tetapi metrik IOPS dan throughput rata-rata mungkin melaporkan bahwa Anda mendorong kinerja yang lebih rendah daripada IOPS atau batas throughput yang disediakan volume Anda. Untuk mengidentifikasi apakah volume Anda mengalami peningkatan kinerja dalam satu menit tertentu, Anda dapat menggunakan `VolumeThroughputExceededCheck` metrik `VolumeIOPSExceededCheck` dan. Anda dapat memantau metrik ini untuk menentukan apakah beban kerja Anda secara konsisten berusaha mendorong IOPS atau throughput yang lebih besar dari kinerja volume yang disediakan dalam satu menit tertentu. Jika IOPS yang didorong untuk setiap detik dalam satu menit secara konsisten melebihi kinerja IOPS yang disediakan volume Anda, metrik akan kembali. `VolumeIOPSExceededCheck` `1` Jika throughput yang didorong untuk setiap detik dalam satu menit secara konsisten melebihi kinerja throughput yang disediakan volume Anda, `VolumeThroughputExceededCheck` metrik akan kembali`1`. Jika IOPS dan throughput yang didorong berada dalam kinerja yang disediakan volume Anda, metrik akan kembali. `0`

Jika aplikasi Anda membutuhkan jumlah IOPS yang lebih besar daripada yang dapat diberikan volume, Anda harus mempertimbangkan untuk menggunakan salah satu dari berikut ini:

- Volume `gp3`, `io2`, atau `io1` yang disediakan dengan IOPS yang cukup untuk mencapai latensi yang diperlukan
- Volume `gp2` yang lebih besar yang memberikan performa IOPS dasar yang cukup

HDD-backed `st1`dan `sc1` volume dirancang untuk berkinerja terbaik dengan beban kerja yang memanfaatkan ukuran maksimum I/O 1.024 KiB. Untuk menentukan I/O ukuran rata-rata volume Anda, bagi `VolumeWriteBytes` dengan`VolumeWriteOps`. Penghitungan yang sama berlaku untuk membaca operasi. Jika I/O ukuran rata-rata di bawah 64 KiB, meningkatkan ukuran I/O operasi yang dikirim ke `sc1` volume `st1` atau akan meningkatkan kinerja.

Pantau keseimbangan bucket burst untuk volume **`gp2`**, **`st1`**, dan **`sc1`**

`BurstBalance` menampilkan saldo bucket lonjakan untuk volume `gp2`, `st1`, dan `sc1` sebagai persentase dari saldo yang tersisa. Saat bucket burst habis, volume I/O (untuk `gp2` volume) atau throughput volume (untuk `st1` dan `sc1` volume) dibatasi ke baseline. Periksa nilai `BurstBalance` untuk menentukan apakah volume Anda dipacu karena alasan ini. Untuk daftar lengkap metrik Amazon EBS yang tersedia, lihat CloudWatch Metrik Amazon untuk Amazon EBS dan metrik Amazon EBS untuk Nitro-based instans.

Memantau statistik I/O kinerja waktu nyata

Anda dapat mengakses statistik performa terperinci real-time untuk volume Amazon EBS yang dilampirkan ke instans Nitro-based Amazon EC2.

Anda dapat menggabungkan statistik ini untuk mendapatkan latensi rata-rata dan IOPS, atau untuk memeriksa apakah I/O operasi selesai. Anda juga dapat melihat jumlah total waktu aplikasi Anda telah melampaui batas IOPS atau throughput yang disediakan volume EBS atau instans terlampir. Dengan melacak peningkatan statistik ini dari waktu ke waktu, Anda dapat mengidentifikasi apakah Anda perlu meningkatkan IOPS atau batas throughput yang disediakan untuk mengoptimalkan kinerja aplikasi Anda. Statistik kinerja terperinci juga mencakup histogram untuk I/O operasi baca dan tulis, yang memberikan distribusi I/O latensi Anda dengan melacak jumlah total I/O operasi yang diselesaikan dalam pita latensi.

Untuk informasi selengkapnya, lihat Statistik kinerja terperinci Amazon EBS.

Sumber daya terkait

Untuk informasi selengkapnya tentang I/O karakteristik Amazon EBS, lihat presentasi re:Invent berikut: Amazon EBS: Designing for Performance.

Konfigurasi Amazon EBS dan RAID

Dengan Amazon EBS, Anda dapat menggunakan salah satu konfigurasi RAID standar yang dapat Anda gunakan dengan server bare metal tradisional, selama konfigurasi RAID tersebut didukung oleh sistem operasi untuk instans Anda. Ini karena semua RAID dilakukan di tingkat perangkat lunak.

Data volume Amazon EBS direplikasi di banyak server di Zona Ketersediaan untuk mencegah hilangnya data akibat kegagalan komponen apa pun. Replikasi ini membuat volume Amazon EBS mencapai 10 kali lebih dapat diandalkan dibandingkan drive disk yang biasa. Untuk informasi selengkapnya, lihat fitur https://aws.amazon.com/ebs/features/ Amazon EBS.

Daftar Isi

- Opsi konfigurasi RAID
- Membuat array RAID 0
- Buat snapshot volume dalam suatu array RAID

Opsi konfigurasi RAID

Membuat rangkaian RAID 0 memungkinkan Anda mencapai tingkat performa yang lebih tinggi untuk sistem file daripada yang dapat Anda berikan dalam satu volume Amazon EBS. Gunakan RAID 0 ketika I/O kinerja adalah yang paling penting. Dengan RAID 0, I/O didistribusikan di seluruh volume dalam garis. Jika Anda menambah volume, Anda mendapatkan penambahan langsung dari throughput dan IOPS. Namun, perlu diingat bahwa performa stripe terbatas pada volume beperforma terburuk di set, dan bahwa hilangnya satu volume dalam hasil set dalam kehilangan data lengkap untuk array.

Ukuran yang dihasilkan dari array RAID 0 adalah jumlah ukuran volume didalamnya, dan bandwidth-nya adalah jumlah bandwidth yang tersedia dari volume di dalamnya. Misalnya, dua `io1` volume 500 GiB dengan 4.000 IOPS yang disediakan masing-masing membuat array 1.000 GiB RAID 0 dengan bandwidth yang tersedia sebesar 8.000 IOPS dan 1.000 throughput. MiB/s

> **Important**
>
> RAID 5 dan RAID 6 tidak disarankan untuk Amazon EBS karena operasi tulis paritas pada mode RAID ini menghabiskan beberapa IOPS yang tersedia untuk volume Anda. Bergantung pada konfigurasi array RAID Anda, mode RAID ini menyediakan IOPS yang dapat digunakan 20-30% lebih sedikit dibandingkan konfigurasi RAID 0. Peningkatan biaya adalah faktor yang juga memengaruhi mode RAID ini; ketika menggunakan ukuran volume dan kecepatan yang sama, array RAID 0 2 volume dapat melampaui array RAID 6 4 volume yang berbiaya dua kali lebih banyak.
>
> RAID 1 juga tidak disarankan untuk digunakan dengan Amazon EBS. RAID 1 memerlukan lebih banyak Amazon EC2 untuk bandwidth Amazon EBS daripada konfigurasi non-RAID karena data ditulis ke beberapa volume secara bersamaan. Selain itu, RAID 1 tidak memberikan peningkatan performa tulis.

Membuat array RAID 0

Gunakan prosedur berikut untuk membuat rangkaian RAID 0.

Pertimbangan-pertimbangan

- Sebelum Anda melakukan prosedur ini, Anda harus memutuskan seberapa besar array RAID 0 Anda seharusnya dan berapa banyak IOPS yang akan disediakan.

- Buat ukuran yang sama dan nilai performa IOPS untuk array Anda. Pastikan Anda tidak membuat array yang melebihi bandwidth yang tersedia dari instans EC2 Anda.
- Anda harus menghindari boot dari volume RAID. Jika salah satu perangkat gagal, Anda mungkin tidak dapat mem-boot sistem operasi.

Instans Linux

Untuk membuat array RAID 0 di Linux

1. Buat volume Amazon EBS untuk array Anda. Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.
2. Lampirkan volume Amazon EBS ke instans di mana Anda ingin melakukan hosting array tersebut. Untuk informasi selengkapnya, lihat Melampirkan volume Amazon EBS ke instans Amazon EC2.
3. Gunakan perintah mdadm untuk membuat perangkat RAID logis dari volume Amazon EBS yang baru dipasang. Ganti jumlah volume dalam array Anda untuk *number_of_volumes* dan nama perangkat untuk setiap volume dalam array (seperti `/dev/xvdf`) untuk*device_name*. Anda juga dapat mengganti *MY_RAID* dengan nama unik Anda sendiri untuk array.

   > **Note**
   >
   > Anda dapat mencantumkan perangkat di instans Anda dengan perintah lsblk untuk menemukan nama perangkat.

   Untuk membuat rangkaian RAID 0, jalankan perintah berikut (perhatikan opsi `--level=0` untuk membuat rangkaian strip):

   ```
   [ec2-user ~]$ sudo mdadm --create --verbose /dev/md0 --level=0 --name=MY_RAID --raid-devices=number_of_volumes device_name1 device_name2
   ```

   > **Tip**
   >
   > Jika Anda mendapatkan kesalahan `mdadm: command not found`, gunakan perintah berikut untuk menginstal mdadm: `sudo yum install mdadm`.

4. Berikan waktu untuk array RAID untuk diinisialisasi dan disinkronkan. Anda dapat melacak kemajuan operasi ini dengan perintah berikut:

```
[ec2-user ~]$ sudo cat /proc/mdstat
```

Berikut ini adalah output contoh:

```
Personalities : [raid0]
md0 : active raid0 xvdc[1] xvdb[0]
      41910272 blocks super 1.2 512k chunks

unused devices: <none>
```

Secara umum, Anda dapat menampilkan informasi terperinci tentang rangkaian RAID Anda dengan perintah berikut:

```
[ec2-user ~]$ sudo mdadm --detail /dev/md0
```

Berikut ini adalah output contoh:

```
/dev/md0:
           Version : 1.2
     Creation Time : Wed May 19 11:12:56 2021
        Raid Level : raid0
        Array Size : 41910272 (39.97 GiB 42.92 GB)
      Raid Devices : 2
     Total Devices : 2
       Persistence : Superblock is persistent

       Update Time : Wed May 19 11:12:56 2021
             State : clean
    Active Devices : 2
   Working Devices : 2
    Failed Devices : 0
     Spare Devices : 0

        Chunk Size : 512K

Consistency Policy : none

              Name : MY_RAID
```

```
              UUID : 646aa723:db31bbc7:13c43daf:d5c51e0c
            Events : 0

    Number   Major   Minor   RaidDevice State
       0     202       16        0      active sync   /dev/sdb
       1     202       32        1      active sync   /dev/sdc
```

5. Buat sistem file di array RAID Anda, dan berikan label pada sistem file untuk digunakan saat Anda memasangnya nanti. Misalnya, untuk membuat sistem file ext4 dengan label*MY_RAID*, jalankan perintah berikut:

```
[ec2-user ~]$ sudo mkfs.ext4 -L MY_RAID /dev/md0
```

   Bergantung pada persyaratan aplikasi Anda atau batasan sistem operasi Anda, Anda dapat menggunakan jenis sistem file yang berbeda, seperti ext3 atau XFS (baca dokumentasi sistem file Anda untuk perintah pembuatan sistem file terkait).

6. Untuk memastikan bahwa array RAID dipasang kembali secara otomatis saat boot, buat file konfigurasi untuk berisi informasi RAID:

```
[ec2-user ~]$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf
```

   > **Note**
   >
   > Jika Anda menggunakan distribusi Linux selain Amazon Linux, Anda mungkin perlu memodifikasi perintah ini. Misalnya, Anda mungkin perlu menempatkan file di lokasi yang berbeda, atau Anda mungkin perlu menambahkan parameter `--examine`. Untuk informasi selengkapnya, jalankan man mdadm.conf di instans Linux Anda.

7. Buat image ramdisk baru untuk memuat modul perangkat blok sebelumnya dengan benar untuk konfigurasi RAID Anda yang baru:

```
[ec2-user ~]$ sudo dracut -H -f /boot/initramfs-$(uname -r).img $(uname -r)
```

8. Buat titik pemasangan untuk array RAID Anda.

```
[ec2-user ~]$ sudo mkdir -p /mnt/raid
```

9. Terakhir, pasang perangkat RAID pada titik pemasangan yang Anda buat:

```
[ec2-user ~]$ sudo mount LABEL=MY_RAID /mnt/raid
```

Perangkat RAID Anda sekarang siap digunakan.

10. (Opsional) Untuk memasang volume Amazon EBS ini pada setiap boot ulang sistem, tambahkan entri untuk perangkat ke file `/etc/fstab` Anda.

    a. Buat cadangan dari file `/etc/fstab` Anda yang dapat digunakan jika Anda secara tidak sengaja menghancurkan atau menghapus file ini saat mengedit.

    ```
    [ec2-user ~]$ sudo cp /etc/fstab /etc/fstab.orig
    ```

    b. Buka file `/etc/fstab` menggunakan editor teks favorit Anda seperti nano atau vim.

    c. Berikan komentar untuk setiap baris yang dimulai dengan "`UUID=`" dan, di akhir file, tambahkan baris baru untuk volume RAID Anda menggunakan format berikut:

    ```
    device_label mount_point file_system_type fs_mntops fs_freq fs_passno
    ```

    Tiga kolom terakhir pada baris ini adalah opsi pemasangan sistem file, frekuensi pembuangan sistem file, dan urutan pemeriksaan sistem file yang dilakukan pada waktu booting. Jika Anda tidak tahu nilai-nilai ini, gunakan nilai-nilai di bawah ini untuk mereka (`defaults,nofail 0 2`). Untuk informasi selengkapnya tentang entri `/etc/fstab`, lihat halaman manual fstab (dengan memasukkan man fstab pada baris perintah). Misalnya, untuk memasang sistem file ext4 di perangkat dengan label MY_RAID di titik pemasangan `/mnt/raid`, tambahkan entri berikut ke `/etc/fstab`.

    > **Note**
    >
    > Jika Anda ingin melakukan boot instans tanpa volume terlampir ini (misalnya, sehingga volume ini dapat berpindah bolak-balik antar instans yang berbeda), Anda harus menambahkan opsi pemasangan `nofail` yang memungkinkan instans melakukan boot meskipun terdapat kesalahan dalam pemasangan volume. Derivatif Debian, seperti Ubuntu, juga harus menambah opsi pemasangan `nobootwait`.

    ```
    LABEL=MY_RAID       /mnt/raid   ext4    defaults,nofail        0       2
    ```

d. Setelah Anda menambahkan entri baru ke `/etc/fstab`, Anda perlu memeriksa bahwa entri Anda berfungsi. Jalankan perintah sudo mount -a untuk memasang semua sistem file di `/etc/fstab`.

```
[ec2-user ~]$ sudo mount -a
```

Jika perintah sebelumnya tidak menghasilkan kesalahan, file `/etc/fstab` Anda baik-baik saja dan sistem file Anda akan terpasang secara otomatis di boot berikutnya. Jika perintah tidak menyebabkan kesalahan apa pun, AMI kesalahan tersebut dan coba koreksi `/etc/fstab`.

> **Warning**
>
> Kesalahan dalam file `/etc/fstab` dapat membuat sistem tidak dapat dibooting. Jangan mematikan sistem yang memiliki kesalahan di file `/etc/fstab` Anda.

e. (Opsional) Jika Anda tidak yakin cara mengoreksi kesalahan `/etc/fstab`, Anda selalu dapat memulihkan file `/etc/fstab` cadangan dengan perintah berikut.

```
[ec2-user ~]$ sudo mv /etc/fstab.orig /etc/fstab
```

Instans Windows

Untuk membuat array RAID 0 di Windows

1. Buat volume Amazon EBS untuk array Anda. Untuk informasi selengkapnya, lihat Buat volume Amazon EBS.
2. Lampirkan volume Amazon EBS ke instans di mana Anda ingin melakukan hosting array tersebut. Untuk informasi selengkapnya, lihat Melampirkan volume Amazon EBS ke instans Amazon EC2.
3. Hubungkan ke instans Windows Anda. Untuk informasi selengkapnya, lihat Terhubung ke instans Windows Anda.
4. Buka jendela perintah dan ketikkan perintah diskpart.

```
diskpart

Microsoft DiskPart version 6.1.7601
```

```
Copyright (C) 1999-2008 Microsoft Corporation.
On computer: WIN-BM6QPPL51CO
```

5. Pada perintah DISKPART, buat daftar disk yang tersedia dengan perintah berikut.

```
DISKPART> list disk

  Disk ###  Status         Size     Free     Dyn  Gpt
  --------  -------------  -------  -------  ---  ---
  Disk 0    Online           30 GB      0 B
  Disk 1    Online            8 GB      0 B
  Disk 2    Online            8 GB      0 B
```

   Identifikasi disk yang ingin Anda gunakan dalam array Anda dan catat nomor disknya.

6. Setiap disk yang ingin Anda gunakan dalam array Anda harus berupa disk dinamik online yang tidak berisi volume apa pun yang ada. Gunakan langkah-langkah berikut untuk mengonversi disk dasar menjadi disk dinamik dan untuk menghapus volume yang ada.

   a. Pilih disk yang ingin Anda gunakan dalam array Anda dengan perintah berikut, ganti *n* dengan nomor disk Anda.

```
DISKPART> select disk n

Disk n is now the selected disk.
```

   b. Jika disk yang dipilih tercantum sebagai Offline, bawa online dengan menjalankan perintah online disk.

   c. Jika disk yang dipilih tidak memiliki tanda bintang dalam kolom Dyn di output perintah list disk sebelumnya, Anda perlu mengonversinya ke disk dinamis.

```
DISKPART> convert dynamic
```

   > **Note**
   >
   > Jika Anda menerima kesalahan bahwa disk tidak dapat ditulis, Anda dapat menghapus tanda hanya-baca dengan perintah ATTRIBUTE DISK CLEAR READONLY kemudian coba konversi disk dinamis lagi.

   d. Gunakan perintah detail disk untuk memeriksa volume yang ada pada disk yang dipilih.

```
DISKPART> detail disk

XENSRC PVDISK SCSI Disk Device
Disk ID: 2D8BF659
Type   : SCSI
Status : Online
Path   : 0
Target : 1
LUN ID : 0
Location Path : PCIROOT(0)#PCI(0300)#SCSI(P00T01L00)
Current Read-only State : No
Read-only  : No
Boot Disk  : No
Pagefile Disk  : No
Hibernation File Disk  : No
Crashdump Disk  : No
Clustered Disk  : No

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 2     D   NEW VOLUME   FAT32  Simple      8189 MB  Healthy
```

Perhatikan nomor volume pada disk. Dalam contoh ini, jumlah volume adalah 2. Jika tidak ada volume, Anda dapat melewati langkah berikutnya.

e. (Hanya diperlukan jika volume diidentifikasi di langkah sebelumnya) Pilih dan hapus volume yang ada pada disk yang Anda identifikasi di langkah sebelumnya.

> **⚠ Warning**
>
> Ini menghancurkan semua data yang ada pada volume.

i. Pilih volume, ganti *n* dengan nomor volume Anda.

```
DISKPART> select volume n
Volume n is the selected volume.
```

ii. Hapus volume.

```
DISKPART> delete volume
```

```
DiskPart successfully deleted the volume.
```

iii. Ulangi sublangkah ini untuk setiap volume yang perlu dihapus pada disk yang dipilih.

f. Ulangi Step 6 untuk setiap disk yang ingin Anda gunakan dalam array.

7. Verifikasi bahwa disk yang ingin Anda gunakan sekarang adalah dinamik. Dalam hal ini, contoh ini menggunakan disk 1 dan 2 untuk volume RAID.

```
DISKPART> list disk

  Disk ###  Status         Size     Free     Dyn  Gpt
  --------  -------------  -------  -------  ---  ---
  Disk 0    Online           30 GB      0 B
  Disk 1    Online            8 GB      0 B   *
  Disk 2    Online            8 GB      0 B   *
```

8. Buat array raid Anda. Pada Windows, volume RAID 0 disebut sebagai volume striped.

   Untuk membuat array volume striped pada disk 1 dan 2, gunakan perintah berikut (perhatikan opsi `stripe` untuk membuat strip array):

```
DISKPART> create volume stripe disk=1,2
DiskPart successfully created the volume.
```

9. Verifikasi volume baru Anda.

```
DISKPART> list volume

  DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     C                NTFS   Partition     29 GB  Healthy    System
  Volume 1                      RAW    Stripe        15 GB  Healthy
```

   Perhatikan bahwa kolom Type sekarang menunjukkan bahwa Volume 1 adalah volume `stripe`.

10. Pilih dan format volume Anda sehingga Anda dapat mulai menggunakannya.

    a. Pilih volume yang ingin Anda format, ganti *n* dengan nomor volume Anda.

```
DISKPART> select volume n
```

```
Volume n is the selected volume.
```

b. Format volume.

> **Note**
>
> Untuk melakukan format penuh, hapus opsi `quick`.

```
DISKPART> format quick recommended label="My new volume"

  100 percent completed

DiskPart successfully formatted the volume.
```

c. Tetapkan huruf drive yang tersedia untuk volume Anda.

```
DISKPART> assign letter f

DiskPart successfully assigned the drive letter or mount point.
```

Volume baru Anda sekarang siap digunakan.

Buat snapshot volume dalam suatu array RAID

Jika Anda ingin mencadangkan data pada volume EBS dalam array RAID menggunakan snapshot, Anda harus memastikan bahwa snapshot konsisten. Ini karena snapshot volume ini dibuat secara independen. Untuk memulihkan volume EBS dalam array RAID dari snapshot yang tidak selaras akan menurunkan integritas dari array.

Untuk membuat kumpulan snapshot yang konsisten untuk array RAID Anda, gunakan snapshot multi-volume EBS. Multi-volume snapshot memungkinkan Anda mengambil snapshot point-in-time, data terkoordinasi, dan konsisten crash di beberapa volume EBS yang terpasang ke instans EC2. Anda tidak perlu menghentikan instans untuk berkoordinasi antar volume untuk memastikan konsistensi, karena snapshot diambil secara otomatis di beberapa volume EBS. Untuk informasi selengkapnya, lihat langkah-langkah untuk membuat snapshot multi-volume di bawah Membuat snapshot Amazon EBS.

Tolok ukur volume Amazon EBS

Anda dapat menguji kinerja volume Amazon EBS dengan mensimulasikan beban kerja I/O . Prosesnya adalah sebagai berikut:

1. Luncurkan sebuah EBS-optimized instance.
2. Buat volume EBS baru.
3. Lampirkan volume ke EBS-optimized instance Anda.
4. Konfigurasikan dan pasang perangkat blok.
5. Instal alat untuk mengukur I/O kinerja.
6. Tolok ukur I/O kinerja volume Anda.
7. Hapus volume Anda dan akhiri instans Anda sehingga Anda tidak terus membebankan biaya.

> **Important**
>
> Beberapa prosedur mengakibatkan penghancuran data yang ada pada volume EBS yang menjadi patokan. Prosedur tolok ukur dimaksudkan untuk digunakan pada volume yang dibuat khusus untuk tujuan pengujian, bukan volume produksi.

Siapkan instans Anda

Untuk mendapatkan kinerja optimal dari volume EBS, sebaiknya gunakan EBS-optimized instance. EBS-optimized instans memberikan bandwidth khusus antara Amazon EC2 dan Amazon EBS, dengan spesifikasi tergantung pada jenis instans.

Untuk membuat EBS-optimized instance, pilih Lun curkan sebagai EBS-optimized instance saat meluncurkan instans menggunakan konsol Amazon EC2, atau tentukan --ebs-optimized saat menggunakan baris perintah. Pastikan Anda memilih jenis instance yang mendukung opsi ini.

Menyiapkan volume SSD IOPS yang Tersedia atau SSD Tujuan Umum

Untuk menciptakan volume SSD IOPS yang Tersedia (`io1` dan `io2`) atau SSD Tujuan Umum (`gp2` dan `gp3`) menggunakan konsol Amazon EC2, untuk Tipe volume, pilih SSD IOPS yang Tersedia (io1), SSD IOPS yang Tersedia (io2), SSD Tujuan Umum (gp2), atau SSD Tujuan Umum (gp3). Di

baris perintah, tentukan`io1`, `io2`, `gp2`, atau `gp3` untuk parameter --volume-type. Untuk`io1`,`io2`, dan `gp3` volume, tentukan jumlah I/O operasi per detik (IOPS) untuk --iops parameter. Untuk informasi selengkapnya, lihat Tipe volume Amazon EBS dan Buat volume Amazon EBS.

(Hanya instans Linux) Untuk pengujian contoh, kami sarankan Anda membuat array RAID 0 dengan 6 volume, yang menawarkan tingkat kinerja tinggi. Karena Anda dikenai biaya berdasarkan gigabita yang disediakan (dan jumlah IOPS yang Tersedia untuk volume io1, io2, dan gp3), bukan jumlah volume, tidak ada biaya tambahan untuk membuat beberapa volume yang lebih kecil dan menggunakannya untuk membuat set stripe. Jika Anda menggunakan Oracle Orion untuk mengukur volume Anda, Oracle Orion dapat melakukan simulasi striping dengan cara yang sama seperti yang dilakukan Oracle ASM, jadi sebaiknya biarkan Orion yang melakukan striping. Jika Anda menggunakan alat tolok ukur yang berbeda, Anda perlu membuat volume sendiri.

Untuk informasi selengkapnya tentang cara membuat array RAID 0, lihatMembuat array RAID 0.

Menyiapkan volume HDD yang Dioptimalkan Th **roughput (st1**) atau Cold HDD (**sc1**)

Untuk membuat volume `st1`, pilih HDD Throughput Dioptimalkan saat membuat volume menggunakan konsol Amazon EC2, atau tentukan --type **`st1`** saat menggunakan baris perintah. Untuk membuat volume `sc1`, pilih HDD Throughput Dioptimalkan saat membuat volume menggunakan konsol Amazon EC2, atau tentukan --type **`sc1`** saat menggunakan baris perintah. Untuk informasi tentang pembuatan volume EBS, lihat Buat volume Amazon EBS. Untuk informasi tentang memasang volume ini ke instans Anda, lihat Melampirkan volume Amazon EBS ke instans Amazon EC2.

(Hanya instans Linux) AWS menyediakan template JSON untuk digunakan CloudFormation yang menyederhanakan prosedur penyiapan ini. Akses template dan simpan sebagai file JSON. CloudFormation memungkinkan Anda untuk mengkonfigurasi kunci SSH Anda sendiri dan menawarkan cara yang lebih mudah untuk mengatur lingkungan pengujian kinerja untuk mengevaluasi `st1` volume. Templat membuat instans generasi saat ini dan 2 TiB volume `st1`, dan memasangkan volume ke instans pada `/dev/xvdf`.

(Hanya instans Linux) Untuk membuat volume HDD menggunakan template

1. Buka CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
2. Pilih Buat tumpukan.
3. Pilih Unggah Templat ke Amazon S3 dan pilih templat JSON yang Anda dapatkan sebelumnya.

4. Berikan nama tumpukan Anda seperti “ebs-perf-testing”, dan pilih tipe instans (defaultnya adalah r3.8xlarge) dan kunci SSH.
5. Pilih Selanjutnya dua kali, lalu pilih Buat Tumpukan.
6. Setelah status untuk tumpukan baru Anda berpindah dari CREATE_IN_PROGRESS ke COMPLETE, pilih Output untuk mendapatkan entri DNS publik untuk instans baru Anda, yang akan memiliki volume `st1` 2 TiB yang terlampir padanya.
7. Terhubung menggunakan SSH ke tumpukan baru Anda sebagai pengguna **`ec2-user`**, dengan nama host yang diperoleh dari entri DNS di langkah sebelumnya.
8. Lanjut ke Pasang alat tolok ukur.

Pasang alat tolok ukur

Tabel berikut mencantumkan beberapa alat yang mungkin dapat Anda gunakan untuk membandingkan kinerja volume EBS.

Instans Linux

Alat	Deskripsi

fio	Untuk I/O kinerja benchmarking. (Perhatikan bahwa fio memiliki ketergant ungan pada `libaio-devel` .)
	
	Untuk menginstal fio di Amazon Linux, jalankan perintah berikut:
	
	`$ sudo yum install -y fio`
	
	Untuk memasang fio di Ubuntu, jalankan perintah berikut:
	
	`sudo apt-get install -y fio`

Alat Kalibrasi Orion Oracle	Untuk mengkalibrasi I/O kinerja sistem penyimpanan yang akan digunakan dengan database Oracle.

Instans Windows

Alat	Deskripsi
DiskSpd	DiskSpd adalah alat kinerja penyimpanan dari tim rekayasa Infrastruktur Windows, Windows Server, dan Cloud Server di Microsoft. Ini tersedia untuk diunduh di https://github.com/Microsoft/diskspd/releases. Setelah Anda mengunduh file `diskspd.exe` yang dapat dijalankan, buka command promt dengan wewenang administratif (dengan memilih "Run as Administrator"), dan kemudian navigasi ke direktori tempat Anda menyalin file `diskspd.exe` . Salin yang diinginkan file `diskspd.exe` yang dapat dieksekusi dari folder executable yang sesuai (`amd64fre`, `armfre` atau `x86fre`) ke jalur yang singkat dan sederhana seperti `C:\DiskSpd` . Dalam kebanyakan kasus, Anda akan menginginkan versi 64-bit DiskSpd dari `amd64fre` folder. Kode sumber untuk DiskSpd di-host GitHub di: https://github.com/Microsoft/diskspd.
CrystalDiskMark	CrystalDiskMark adalah perangkat lunak benchmark disk sederhana. Ini tersedia untuk diunduh di https://crystalmark.info/en/software/crystaldiskmark/.

Alat tolok ukur ini mendukung berbagai macam parameter uji. Anda harus menggunakan perintah yang akan mendukung oleh perkiraan beban kerja volume Anda. Perintah yang diberikan di bawah ini dimaksudkan sebagai contoh untuk membantu Anda memulai.

Pilih panjang antrean volume

Memilih panjang antrean volume terbaik berdasarkan beban kerja dan tipe volume Anda.

Panjang antrian pada SSD-backed volume

Untuk menentukan panjang antrian optimal untuk beban kerja pada SSD-backed volume, sebaiknya Anda menargetkan panjang antrian 1 untuk setiap 1000 IOPS yang tersedia (dasar untuk volume SSD Tujuan Umum dan jumlah yang disediakan untuk volume SSD IOPS Tersediakan). Kemudian,

Anda dapat memantau performa aplikasi Anda dan menyesuaikan nilai tersebut berdasarkan kebutuhan aplikasi Anda.

Peningkatan panjang antrean akan bermanfaat hingga Anda mencapai IOPS yang tersedia , throughput, atau panjang antrean sistem optimal, yang saat ini ditetapkan ke 32. Misalnya, volume dengan 3.000 IOPS yang Tersedia harus menargetkan panjang antrean 3. Anda harus bereksperimen mengatur nilai-nilai ini ke atas atau ke bawah untuk melihat apa yang terbaik untuk aplikasi Anda.

Panjang antrian pada HDD-backed volume

Untuk menentukan panjang antrian optimal untuk beban kerja Anda pada HDD-backed volume, sebaiknya Anda menargetkan panjang antrian minimal 4 saat melakukan 1MiB berurutan. I/Os Kemudian, Anda dapat memantau performa aplikasi Anda dan menyesuaikan nilai tersebut berdasarkan kebutuhan aplikasi Anda. Misalnya, `st1` volume 2 TiB dengan throughput burst 500 MiB/s dan IOPS 500 harus menargetkan panjang antrian 4, 8, atau 16 saat melakukan masing-masing 1.024 KiB, 512 KiB, atau 256 KiB berurutan. I/Os Anda harus bereksperimen mengatur nilai-nilai ini ke atas atau ke bawah untuk melihat apa yang terbaik untuk aplikasi Anda.

Nonaktifkan C-states

Sebelum Anda menjalankan benchmarking, Anda harus menonaktifkan prosesor C-states. Core idle sementara di CPU yang didukung dapat memasukkan a C-state untuk menghemat daya. Ketika inti dipanggil untuk melanjutkan pemrosesan, beberapa waktu berlalu sampai inti beroperasi GApenuh. Latensi ini dapat mengganggu rutinitas tolok ukur prosesor . Untuk informasi selengkapnya tentang C-states dan jenis instans EC2 mana yang mendukungnya, lihat Kontrol status prosesor untuk instans EC2 Anda.

Instans Linux

Anda dapat menonaktifkan C-states di Amazon Linux, RHEL, dan CentOS sebagai berikut:

1. Dapatkan jumlah C-states.

```
$ cpupower idle-info | grep "Number of idle states:"
```

2. Nonaktifkan C-states dari c1 ke cN. Idealnya, inti harus berada dalam keadaan c0.

```
$ for i in `seq 1 $((N-1))`; do cpupower idle-set -d $i; done
```

Instans Windows

Anda dapat menonaktifkan C-states pada Windows sebagai berikut:

1. Mas PowerShell uk, dapatkan skema daya aktif saat ini.

```
$current_scheme = powercfg /getactivescheme
```

2. Dapatkan skema daya GUID.

```
(Get-WmiObject -class Win32_PowerPlan -Namespace "root\cimv2\power" -Filter
 "ElementName='High performance'").InstanceID
```

3. Dapatkan pengaturan daya GUID.

```
(Get-WmiObject -class Win32_PowerSetting -Namespace "root\cimv2\power" -Filter
 "ElementName='Processor idle disable'").InstanceID
```

4. Dapatkan pengaturan daya subgrup GUID.

```
(Get-WmiObject -class Win32_PowerSettingSubgroup -Namespace "root\cimv2\power" -
Filter "ElementName='Processor power management'").InstanceID
```

5. N C-states onaktifkan dengan mengatur nilai indeks ke 1. Nilai 0 menunjukkan bahwa C-states dinonaktifkan.

```
powercfg /
setacvalueindex <power_scheme_guid> <power_setting_subgroup_guid> <power_setting_guid>
 1
```

6. Tetapkan skema aktif untuk memastikan pengaturan disimpan.

```
powercfg /setactive <power_scheme_guid>
```

Lakukan benchmarking

Prosedur berikut menjelaskan perintah tolok ukur untuk berbagai tipe volume EBS.

Jalankan perintah berikut pada EBS-optimized instance dengan volume EBS terlampir. Jika volume EBS dibuat dari snapshot, pastikan untuksebelum menetapkan tolok ukur. Untuk informasi selengkapnya, lihat Secara manual menginisialisasi volume setelah pembuatan.

> **Tip**
>
> Anda dapat menggunakan histogram I/O latensi yang disediakan oleh statistik kinerja terperinci EBS untuk membandingkan distribusi I/O kinerja dalam tes benchmarking Anda. Untuk informasi selengkapnya, lihat Statistik kinerja terperinci Amazon EBS.

Setelah selesai menguji volume, lihat topik berikut untuk bantuan pembersihan: Menghapus volume Amazon EBS dan Hentikan instans Anda.

Tolok Ukur Volume SSD IOPS yang Tersedia dan SSD Tujuan Umum

Instans Linux

Jalankan fio pada array RAID 0 yang Anda buat.

Perintah berikut melakukan operasiacak 16 KB.

```
$ sudo fio --directory=/mnt/p_iops_vol0 --ioengine=psync --name fio_test_file --
direct=1 --rw=randwrite --bs=16k --size=1G --numjobs=16 --time_based --runtime=180 --
group_reporting --norandommap
```

Perintah berikut melakukan operasi baca acak 16 KB.

```
$ sudo fio --directory=/mnt/p_iops_vol0 --name fio_test_file --direct=1 --rw=randread
 --bs=16k --size=1G --numjobs=16 --time_based --runtime=180 --group_reporting --
norandommap
```

Untuk informasi selengkapnya tentang penafsiran hasil, lihat tutorial ini: Memeriksa performa IO disk dengan fio.

Instans Windows

Jalankan DiskSpd pada volume yang Anda buat.

Perintah berikut akan menjalankan I/O tes acak 30 detik menggunakan file uji 20GB yang terletak di `C:` drive, dengan rasio tulis 25% dan baca 75%, dan ukuran blok 8K. Ini akan menggunakan delapan utas pekerja, masing-masing dengan empat yang luar biasa I/Os, dan benih nilai entropi tulis 1GB. Hasil uji akan disimpan ke file teks yang disebut `DiskSpeedResults.txt`. Parameter ini mensimulasikan beban kerja SQL Server OLTP.

```
diskspd -b8K -d30 -o4 -t8 -h -r -w25 -L -Z1G -c20G C:\iotest.dat > DiskSpeedResults.txt
```

Untuk informasi selengkapnya tentang penafsiran hasil, lihat tutorial ini: Memeriksa performa IO disk dengan fio.

Benchmark `volume st1 dan` sc1 (instans Linux)

Jalankan fio pada volume `st1` atau `sc1`.

> **Note**
>
> Sebelum menjalankan pengujian ini, atur buffered I/O pada instance Anda seperti yang dijelaskan dalamTingkatkan pembacaan ke depan untuk beban kerja throughput tinggi dan berat baca di st1 dan sc1 (Hanya instans Linux).

Perintah berikut melakukan operasi pembacaan berurutan 1 MiB terhadap perangkat blok `st1` terlampir (misalnya, `/dev/xvdf`):

```
$ sudo fio --filename=/dev/<device> --direct=1 --rw=read --randrepeat=0
 --ioengine=libaio --bs=1024k --iodepth=8 --time_based=1 --runtime=180 --
name=fio_direct_read_test
```

Perintah berikut melakukan operasi pembacaan berurutan 1 MiB terhadap perangkat blok `st1` yang terlampir:

```
$ sudo fio --filename=/dev/<device> --direct=1 --rw=write --randrepeat=0
 --ioengine=libaio --bs=1024k --iodepth=8 --time_based=1 --runtime=180 --
name=fio_direct_write_test
```

Beberapa beban kerja melakukan campuran antara baca berurutan danberurutan ke bagian perangkat blok yang berbeda. Untuk mengukur beban kerja tersebut, kami sarankan agar Anda menggunakan pekerjaan fio untuk membaca serta menggunakan opsi fio `offset_increment` untuk menargetkan lokasi perangkat blok yang berbeda untuk setiap pekerjaan.

Menjalankan beban kerja ini adalah yang lebih rumit dibandingkan dengan beban kerja baca-urut atau tulis-urut. Gunakan editor teks untuk membuat file pekerjaan fio, yang disebut `fio_rw_mix.cfg` dalam contoh ini, yang berisi hal berikut:

```
[global]
clocksource=clock_gettime
randrepeat=0
runtime=180

[sequential-write]
bs=1M
ioengine=libaio
direct=1
iodepth=8
filename=/dev/<device>
do_verify=0
rw=write
rwmixread=0
rwmixwrite=100

[sequential-read]
bs=1M
ioengine=libaio
direct=1
iodepth=8
filename=/dev/<device>
do_verify=0
rw=read
rwmixread=100
rwmixwrite=0
offset=100g
```

Kemudian jalankan perintah berikut:

```
$ sudo fio fio_rw_mix.cfg
```

Untuk informasi lebih lanjut tentang menafsirkan hasil, lihat tutorial ini: M emeriksa I/O kinerja disk dengan fi o.

Beberapa fio pekerjaan untuk langsung I/O, meskipun menggunakan operasi baca atau tulis berurutan, dapat menghasilkan throughput `st1` dan `sc1` volume yang lebih rendah dari yang diharapkan. Kami menyarankan Anda menggunakan satu I/O pekerjaan langsung dan menggunakan `iodepth` parameter untuk mengontrol jumlah I/O operasi bersamaan.

Mengotomatiskan pencadangan dengan Amazon Data Lifecycle Manager

Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pembuatan, penyimpanan, dan penghapusan snapshot EBS dan EBS-backed AMI. Ketika Anda mengotomatisasi snapshot dan manajemen AMI, hal ini membantu Anda untuk:

- Lindungi data berharga dengan menerapkan jadwal pencadangan rutin.
- Buat AMI terstandardisasi yang dapat dimuat ulang secara berkala.
- Mempertahankan cadangan sebagaimana diwajibkan oleh auditor atau kepatuhan internal.
- Mengurangi biaya penyimpanan dengan menghapus cadangan yang usang.
- Membuat kebijakan cadangan pemulihan bencana yang membuat cadangan data ke Region atau akun terisolasi.

Dikombinasikan dengan Amazon EventBridge dan AWS CloudTrail pemantauan, Amazon Data Lifecycle Manager menyediakan solusi pencadangan lengkap untuk instans Amazon EC2 dan volume EBS tanpa biaya tambahan.

> **Important**
>
> - Amazon Data Lifecycle Manager tidak dapat mengelola snapshot atau AMI yang dibuat dengan cara lain.
> - Amazon Data Lifecycle Manager tidak dapat mengotomatisasi pembuatan, retensi, dan penghapusan AMI yang didukung penyimpanan instans.

Amazon Data Lifecycle Manager dinilai sebagai kemampuan layanan Amazon Elastic Block Store (Amazon EBS). AWS Layanan apa pun dalam Cakupan berdasarkan Program Kepatuhan (FedRAMP, HIPAA BAA, SOC, dll) yang mencantumkan Amazon EBS juga akan berlaku untuk Amazon Data Lifecycle Manager.

Kuota

AWS Akun Anda memiliki kuota berikut terkait dengan Amazon Data Lifecycle Manager:

Deskripsi	Kuota
Kebijakan siklus hidup kustom per Wilayah	100
Kebijakan default untuk snapshot EBS per Wilayah	1
Kebijakan default untuk EBS-backed AMI per Wilayah	1
Tanda per sumber daya	45

Cara kerja Amazon Data Lifecycle Manager

Berikut ini adalah elemen utama dari Amazon Data Lifecycle Manager.

Elemen

- Kebijakan
- Jadwal kebijakan (kebijakan khusus saja)
- Tag sumber daya target (kebijakan khusus saja)
- Snapshot
- EBS-backed AMI
- Tanda Amazon Data Lifecycle Manager

Kebijakan

Dengan Amazon Data Lifecycle Manager, Anda membuat kebijakan untuk menentukan persyaratan pembuatan dan retensi cadangan. Kebijakan ini biasanya menentukan hal berikut:

- Jenis kebijakan — Mendefinisikan jenis sumber daya cadangan yang dikelola kebijakan (snapshot atau EBS-backed AMI).
- Sumber daya target — Menentukan jenis sumber daya yang ditargetkan oleh kebijakan (instans atau volume EBS).

- Frekuensi pembuatan — Menentukan seberapa sering kebijakan berjalan dan membuat snapshot atau AMI.
- Ambang batas retensi — Menentukan berapa lama kebijakan mempertahankan snapshot atau AMI setelah dibuat.
- Tindakan tambahan — Menentukan tindakan tambahan yang harus dilakukan kebijakan, seperti penyalinan, pengarsipan, atau penandaan sumber daya lintas wilayah.

Amazon Data Lifecycle Manager menawarkan kebijakan default dan kebijakan kustom.

Kebijakan default

Kebijakan default mencadangkan semua volume dan instans di Wilayah yang tidak memiliki cadangan terbaru. Anda dapat mengecualikan volume dan instans secara opsional dengan menentukan parameter pengecualian.

Amazon Data Lifecycle Manager mendukung kebijakan default berikut:

- Kebijakan default untuk snapshot EBS — Menargetkan volume dan mengotomatiskan pembuatan, retensi, dan penghapusan snapshot.
- Kebijakan default untuk EBS-backed AMI — Menargetkan instance dan mengotomatiskan pembuatan, penyimpanan, dan deregistrasi AMI. EBS-backed

Anda hanya dapat memiliki satu kebijakan default per jenis sumber daya di setiap akun dan Wilayah AWS .

Kebijakan kustom

Kebijakan kustom menargetkan sumber daya tertentu berdasarkan tanda yang ditetapkan dan mendukung fitur-fitur canggih, seperti pemulihan snapshot cepat, pengarsipan snapshot, penyalinan lintas akun, serta skrip pra dan pasca. Kebijakan kustom dapat mencakup hingga 4 jadwal, di mana setiap jadwal dapat memiliki frekuensi pembuatan sendiri, ambang retensi, dan konfigurasi fitur lanjutan.

Amazon Data Lifecycle Manager mendukung kebijakan kustom berikut ini:

- Kebijakan default untuk snapshot EBS — Menargetkan volume atau instans dan mengotomatiskan pembuatan, retensi, dan penghapusan snapshot.

- EBS-backed Kebijakan AMI — Menargetkan instance dan mengotomatiskan pembuatan, penyimpanan, dan deregistrasi AMI. EBS-backed
- Cross-account kebijakan peristiwa salin — Mengotomatiskan tindakan penyalinan lintas wilayah untuk snapshot yang dibagikan dengan Anda.

Untuk informasi selengkapnya, lihat Kebijakan default Amazon Data Lifecycle Manager vs kebijakan khusus.

Jadwal kebijakan (kebijakan khusus saja)

Jadwal kebijakan menentukan kapan snapshot atau AMI yang dibuat oleh kebijakan. Kebijakan dapat memiliki hingga empat jadwal—satu jadwal wajib, dan hingga tiga jadwal opsional.

Menambahkan beberapa jadwal ke satu kebijakan memungkinkan Anda membuat snapshot atau AMI pada frekuensi yang berbeda menggunakan kebijakan yang sama. Misalnya, Anda dapat membuat satu kebijakan yang membuat snapshot harian, mingguan, bulanan, dan tahunan. Hal ini menghilangkan kebutuhan untuk mengelola beberapa kebijakan.

Untuk setiap jadwal, Anda dapat menentukan frekuensi, pengaturan pemulihan snapshot cepat (hanya kebijakan siklus hidup snapshot), aturan salinan lintas-Wilayah, dan tanda. Tanda yang ditetapkan ke jadwal secara otomatis ditetapkan ke snapshot atau AMI yang dibuat saat jadwal dimulai. Selain itu, Amazon Data Lifecycle Manager secara otomatis menetapkan tanda yang dihasilkan sistem berdasarkan frekuensi jadwal ke setiap snapshot atau AMI.

Setiap jadwal dimulai secara individual didasarkan pada frekuensinya. Jika beberapa jadwal dimulai secara bersamaan, Amazon Data Lifecycle Manager hanya membuat satu snapshot atau AMI dan menerapkan pengaturan retensi jadwal yang memiliki periode penyimpanan tertinggi. Tanda semua jadwal yang dimulai akan diterapkan ke snapshot atau AMI.

- (Kebijakan siklus hidup snapshot saja) Jika lebih dari satu jadwal yang dimulai diaktifkan untuk pemulihan snapshot cepat, snapshot akan diaktifkan untuk pemulihan snapshot cepat di semua Zona Ketersediaan yang ditentukan di semua jadwal yang dimulai. Pengaturan retensi tertinggi untuk jadwal yang dimulai akan digunakan untuk setiap Zona Ketersediaan.
- Jika lebih dari satu jadwal yang dimulai diaktifkan untuk penyalinan lintas-Wilayah, snapshot atau AMI disalin ke semua Wilayah yang ditentukan di semua jadwal yang dimulai. Periode retensi tertinggi dari jadwal yang dimulai diterapkan.

Tag sumber daya target (kebijakan khusus saja)

Kebijakan kustom Amazon Data Lifecycle Manager menggunakan tanda sumber daya untuk mengidentifikasi sumber daya yang akan dicadangkan. Saat membuat kebijakan snapshot atau EBS-backed AMI, Anda dapat menentukan beberapa tag sumber daya target. Semua sumber daya dari tipe tertentu (instans atau volume) yang memiliki setidaknya satu tanda sumber daya target yang ditentukan akan ditargetkan oleh kebijakan. Misalnya, jika Anda membuat kebijakan snapshot yang menargetkan volume dan Anda menentukan `purpose=prod`, `costcenter=prod`, dan `environment=live` sebagai tanda sumber daya target, kebijakan tersebut akan menargetkan semua volume yang memiliki salah satu pasangan nilai kunci tanda tersebut.

Jika Anda ingin menjalankan beberapa kebijakan pada sumber daya, Anda dapat menetapkan beberapa tanda ke sumber daya target, lalu membuat kebijakan terpisah yang masing-masing menargetkan tanda sumber daya tertentu.

Anda tidak dapat menggunakan karakter \ atau = dalam kunci tanda. Tanda sumber daya peka huruf besar dan kecil. Untuk informasi selengkapnya, lihat Menandai sumber daya Anda.

Snapshot

Snapshot adalah sarana utama untuk mencadangkan data dari volume EBS Anda. Untuk menghemat biaya penyimpanan, snapshot berikutnya bersifat bertahap, hanya berisi data volume yang berubah sejak snapshot sebelumnya. Ketika Anda menghapus satu snapshot dalam seri snapshot untuk volume, hanya data yang unik untuk snapshot itu yang dihapus. Sisa riwayat volume yang ditangkap dipertahankan. Untuk informasi selengkapnya, lihat Snapshot Amazon EBS.

EBS-backed AMI

Amazon Machine Image (AMI) menyediakan informasi yang diperlukan untuk meluncurkan sebuah instans. Anda dapat meluncurkan beberapa instans dari AMI tunggal ketika Anda memerlukan beberapa instans dengan konfigurasi yang sama. Amazon Data Lifecycle Manager EBS-backed hanya mendukung AMI. EBS-backed AMI menyertakan snapshot untuk setiap volume EBS yang dilampirkan ke instance sumber. Untuk informasi selengkapnya, lihat Amazon Machine Images (AMI).

Tanda Amazon Data Lifecycle Manager

Amazon Data Lifecycle Manager menerapkan tanda berikut pada semua snapshot dan AMI yang dibuat oleh kebijakan, untuk membedakannya dari snapshot dan AMI yang dibuat dengan cara lain:

- `aws:dlm:lifecycle-policy-id`
- `aws:dlm:lifecycle-schedule-name`
- `aws:dlm:expirationTime` — Untuk snapshot yang dibuat oleh jadwal berbasis usia. Menunjukkan kapan snapshot akan dihapus dari tingkat standar.
- `dlm:managed`
- `aws:dlm:archived` — Untuk snapshot yang diarsipkan berdasarkan jadwal.
- `aws:dlm:pre-script` — Untuk snapshot yang dibuat dengan skrip pra.
- `aws:dlm:post-script` — Untuk snapshot yang dibuat dengan skrip pasca.

Anda juga dapat menentukan tanda kustom yang akan diterapkan ke snapshot dan AMI pada saat pembuatan. Anda tidak dapat menggunakan karakter \ atau = dalam kunci tanda.

Tanda target yang digunakan Amazon Data Lifecycle Manager untuk mengaitkan volume dengan kebijakan snapshot dapat secara opsional diterapkan pada snapshot yang dibuat oleh kebijakan. Demikian pula, tanda target yang digunakan untuk mengaitkan instans dengan kebijakan AMI dapat secara opsional diterapkan ke AMI yang dibuat oleh kebijakan.

Kebijakan default Amazon Data Lifecycle Manager vs kebijakan khusus

Bagian ini membandingkan kebijakan default dan kebijakan kustom dan menyoroti persamaan dan perbedaannya.

Topik

- Perbandingan kebijakan snapshot EBS
- EBS-backed Perbandingan kebijakan AMI

Perbandingan kebijakan snapshot EBS

Tabel berikut menyoroti perbedaan antara kebijakan default untuk snapshot EBS dan kebijakan snapshot EBS kustom.

Fitur	Kebijakan default untuk snapshot EBS	Kebijakan snapshot EBS kustom
Sumber daya cadangan terkelola	Snapshot EBS	Snapshot EBS
Jenis sumber daya target	Volume	Volume atau instans
Penargetan sumber daya	Menargetkan semua volume di Wilayah yang tidak memiliki snapshot terbaru. Anda dapat menentuka n parameter pengecualian untuk mengecualikan instans tertentu.	Menargetkan hanya volume atau instans yang memiliki tanda tertentu.
Parameter pengecualian	Ya, dapat mengecualikan volume boot, tipe volume tertentu, dan volume dengan tanda tertentu.	Ya, dapat mengecualikan volume dan volume boot dengan tanda tertentu saat menargetkan instans.
Dukungan AWS Outposts	Tidak	Ya
Mendukung beberapa jadwal	Tidak	Ya, hingga 4 jadwal per kebijakan
Jenis retensi yang didukung	Age-based retensi saja	Age-based dan retensi berbasis hitungan
Frekuensi pembuatan snapshot	Setiap 1 hingga 7 hari.	Frekuensi harian, mingguan, bulanan, tahunan, atau kustom menggunakan ekspresi cron.
Retensi snapshot	2 hingga 14 hari.	Hingga 1000 snapshot (berbasis hitungan) atau hingga 100 tahun (berbasis usia).
Mendukung snapshot yang	Tidak	Ya, menggunakan skrip pra dan pasca

Fitur	Kebijakan default untuk snapshot EBS	Kebijakan snapshot EBS kustom
konsisten dengan aplikasi		
Mendukung pengarsipan snapshot	Tidak	Ya
Mendukung pemulihan snapshot cepat	Tidak	Ya
Mendukung penyalinan lintas Wilayah	Ya, dengan pengaturan default [1]	Ya, dengan pengaturan kustom
Mendukung berbagi lintas akun	Tidak	Ya
Mendukung penghapusan yang diperpanj ang [2]	Ya	Tidak

[1] Untuk kebijakan default:

- Anda tidak dapat menyalin tanda ke salinan lintas wilayah.
- Salinan menggunakan periode retensi yang sama dengan snapshot sumber.
- Salinan mendapatkan status enkripsi yang sama dengan snapshot sumber. Jika Wilayah tujuan diaktifkan untuk enkripsi secara default, salinan selalu dienkripsi, bahkan jika snapshot sumber tidak dienkripsi. Salinan selalu dienkripsi dengan kunci KMS default untuk Wilayah tujuan.

[2] Untuk kebijakan default dan kustom:

- Jika instans atau volume target dihapus, Amazon Data Lifecycle Manager terus menghapus snapshot hingga, tetapi tidak termasuk, snapshot terakhir berdasarkan periode retensi. Untuk kebijakan default, Anda dapat memperpanjang penghapusan untuk menyertakan snapshot terakhir.
- Jika kebijakan dihapus atau memasukkan kesalahan atau status dinonaktifkan, Amazon Data Lifecycle Manager berhenti menghapus snapshot. Untuk kebijakan default, Anda dapat memperpanjang penghapusan terus menghapus snapshot, termasuk snapshot terakhir.

EBS-backed Perbandingan kebijakan AMI

Tabel berikut menyoroti perbedaan antara kebijakan default untuk EBS-backed AMI dan kebijakan EBS-backed AMI khusus.

Fitur	Kebijakan default untuk EBS-backed AMI	Kebijakan EBS-backed AMI khusus
Sumber daya cadangan terkelola	EBS-backed AMI	EBS-backed AMI
Jenis sumber daya target	Instans	Instans
Penargetan sumber daya	Menargetkan semua instans di Wilayah yang tidak memiliki AMI terbaru. Anda dapat menentuka n parameter pengecualian untuk mengecualikan instans tertentu.	Menargetkan hanya instans yang memiliki tanda tertentu.
Boot ulang instans sebelum pembuatan AMI	Tidak	Ya
Parameter pengecualian	Ya, dapat mengecualikan instans dengan tanda tertentu.	Tidak

Fitur	Kebijakan default untuk EBS-backed AMI	Kebijakan EBS-backed AMI khusus
Mendukung beberapa jadwal	Tidak	Ya, hingga 4 jadwal per kebijakan.
Frekuensi pembuatan AMI	Setiap 1 hingga 7 hari.	Frekuensi harian, mingguan, bulanan, tahunan, atau kustom menggunakan ekspresi cron.
Jenis retensi yang didukung	Age-based retensi saja.	Age-based dan retensi berbasis hitungan.
Retensi AMI	2 hingga 14 hari.	Hingga 1000 snapshot (berbasis hitungan) atau hingga 100 tahun (berbasis usia).
Mendukung AMI penghentian	Tidak	Ya
Mendukung penyalinan lintas Wilayah	Ya, dengan pengaturan default [1]	Ya, dengan pengaturan kustom
Mendukung penghapusan yang diperpanj ang [2]	Ya	Tidak

[1]Untuk kebijakan default:

- Anda tidak dapat menyalin tanda ke salinan lintas wilayah.
- Salinan menggunakan periode retensi yang sama dengan AMI sumber.
- Salinan mendapatkan status enkripsi yang sama dengan AMI sumber. Jika Wilayah tujuan diaktifkan untuk enkripsi secara default, salinan selalu dienkripsi, bahkan jika AMI sumber tidak dienkripsi. Salinan selalu dienkripsi dengan kunci KMS default untuk Wilayah tujuan.

[2] Untuk kebijakan default dan kustom:

- Jika instans sumber diakhiri, Amazon Data Lifecycle Manager terus membatalkan pendaftaran AMI yang dibuat sebelumnya hingga, tetapi tidak termasuk, yang terakhir berdasarkan periode retensi. Untuk kebijakan default, Anda dapat memperpanjang pembatalan pendaftaran untuk menyertakan AMI terakhir.
- Jika kebijakan dihapus atau memasuki status kesalahan atau dinonaktifkan, Amazon Data Lifecycle Manager berhenti membatalkan pendaftaran AMI. Untuk kebijakan default, Anda dapat memperpanjang pembatalan pendaftaran AMI, termasuk AMI terakhir.

Buat kebijakan default Amazon Data Lifecycle Manager

Untuk membuat EBS-backed AMI periodik dari instans, gunakan kebijakan default untuk EBS-backed AMI. Untuk membuat snapshot dari semua volume terlepas dari status lampirannya, atau jika Anda ingin mengecualikan volume tertentu, gunakan kebijakan default untuk snapshot EBS.

Bagian ini menjelaskan cara membuat kebijakan default.

Topik

- Pertimbangan untuk kebijakan default
- Buat kebijakan default untuk snapshot Amazon EBS
- Buat kebijakan default untuk EBS-backed AMI
- Aktifkan kebijakan default Data Lifecycle Manager di seluruh akun dan Wilayah

Pertimbangan untuk kebijakan default

Ingatlah hal-hal berikut ini saat bekerja dengan kebijakan default:

- Kebijakan default tidak mencadangkan sumber daya target (instans atau volume) yang memiliki cadangan terbaru (snapshot atau AMI). Frekuensi pembuatan menentukan sumber daya yang dicadangkan. Volume atau instans dicadangkan hanya jika snapshot atau AMI terakhirnya lebih tua dari frekuensi pembuatan kebijakan. Misalnya, jika Anda menentukan frekuensi pembuatan 3 hari, kebijakan default untuk snapshot EBS akan membuat snapshot volume hanya jika snapshot terakhirnya lebih tua dari 3 hari.
- Secara default, kebijakan default menargetkan semua instans atau volume di Wilayah, kecuali parameter pengecualian ditentukan.

- Kebijakan default akan membuat set snapshot unik minimum. Misalnya, jika Anda mengaktifkan kebijakan EBS-backed AMI dan kebijakan snapshot EBS, kebijakan snapshot tidak akan menduplikasi snapshot volume yang sudah dicadangkan oleh kebijakan EBS-backed AMI.
- Kebijakan default hanya akan mulai menargetkan sumber daya yang berusia minimal 24 jam.
- Jika Anda menghapus volume atau mengakhiri instans yang ditargetkan oleh kebijakan default, Amazon Data Lifecycle Manager akan terus menghapus cadangan (snapshot atau AMI) yang dibuat sebelumnya sesuai dengan periode retensi hingga, tetapi tidak termasuk, cadangan terakhir. Anda harus menghapus cadangan ini secara manual jika tidak diperlukan.

  Jika ingin Amazon Data Lifecycle Manager menghapus cadangan terakhir, Anda dapat mengaktifkan perpanjang penghapusan.
- Jika kebijakan dihapus atau memasuki status kesalahan atau dinonaktifkan, Amazon Data Lifecycle Manager berhenti menghapus cadangan (snapshot atau AMI) yang dibuat sebelumnya. Jika ingin Amazon Data Lifecycle Manager terus menghapus cadangan, termasuk yang terakhir, Anda harus mengaktifkan perpanjang penghapusan sebelum menghapus kebijakan atau sebelum status kebijakan berubah menjadi dinonaktifkan atau dihapus.
- Saat Anda membuat dan mengaktifkan kebijakan default, Amazon Data Lifecycle Manager secara acak menetapkan sumber daya yang ditargetkan ke jendela waktu empat jam. Sumber daya yang ditargetkan dicadangkan selama jendela yang ditetapkan pada frekuensi pembuatan yang ditentukan. Misalnya, jika kebijakan memiliki frekuensi pembuatan 3 hari, dan sumber daya target ditetapkan ke jendela 12:00 - 16:00, sumber daya tersebut akan dicadangkan antara pukul 12:00 - 16:00 setiap 3 hari.

Buat kebijakan default untuk snapshot Amazon EBS

Prosedur berikut ini menunjukkan cara membuat kebijakan default untuk snapshot EBS.

Console

Untuk membuat kebijakan default untuk snapshot EBS

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Lifecycle Manager lalu pilih Buat kebijakan siklus hidup.
3. Untuk Jenis kebijakan, pilih Kebijakan default, lalu pilih Kebijakan snapshot EBS.
4. Untuk Deskripsi, masukkan deskripsi singkat untuk kebijakan tersebut.
5. Untuk Peran IAM, pilih peran IAM yang memiliki izin untuk mengelola snapshot.

Sebaiknya pilih Default untuk menggunakan peran IAM default yang disediakan oleh Amazon Data Lifecycle Manager. Namun, Anda juga dapat menggunakan peran IAM kustom yang sebelumnya Anda buat.

6. Untuk Frekuensi Pembuatan, tentukan seberapa sering Anda ingin kebijakan berjalan dan membuat snapshot volume Anda.

   Frekuensi yang Anda tentukan juga menentukan volume yang dicadangkan. Kebijakan hanya akan mencadangkan volume yang belum dicadangkan oleh cara lain dalam frekuensi yang ditentukan. Misalnya, jika Anda menentukan frekuensi pembuatan 3 hari, kebijakan hanya akan membuat snapshot volume yang belum dicadangkan dalam 3 hari terakhir.

7. Untuk Periode retensi, tentukan berapa lama Anda ingin kebijakan mempertahankan snapshot yang dibuatnya. Ketika snapshot mencapai ambang retensi, snapshot akan dihapus secara otomatis. Periode retensi harus lebih besar dari atau sama dengan frekuensi pembuatan.

8. (Opsional) Konfigurasikan Parameter pengecualian untuk mengecualikan volume tertentu dari cadangan terjadwal. Volume yang dikecualikan tidak akan dicadangkan saat kebijakan berjalan.

   a. Untuk mengecualikan volume boot, pilih Kecualikan volume boot. Jika Anda mengecualikan volume boot, hanya volume data (non-boot) yang akan dicadangkan oleh kebijakan. Dengan kata lain, kebijakan tidak akan membuat snapshot volume yang dilampirkan ke instans sebagai volume boot.

   b. Untuk mengecualikan tipe volume tertentu, pilih Kecualikan tipe volume tertentu, lalu pilih tipe volume yang akan dikecualikan. Hanya volume dari tipe yang tersisa yang akan didukung oleh kebijakan.

   c. Untuk mengecualikan volume yang memiliki tanda tertentu, pilih Tambahkan tanda, lalu tentukan kunci dan nilai tanda. Kebijakan tidak akan membuat snapshot volume yang memiliki tanda yang ditentukan.

9. (Opsional) Di Pengaturan lanjutan, tentukan tindakan tambahan yang harus dilakukan kebijakan.

   a. Untuk menyalin tanda yang ditetapkan dari volume sumber ke snapshot, pilih Salin tanda dari volume.

   b. Dengan Perpanjang penghapusan dinonaktifkan:

- Jika instans atau volume target dihapus, Amazon Data Lifecycle Manager terus menghapus snapshot hingga, tetapi tidak termasuk, snapshot terakhir berdasarkan periode penyimpanan. Jika Anda ingin Amazon Data Lifecycle Manager menghapus semua snapshot, termasuk yang terakhir, pilih Perpanjang penghapusan.
- Jika kebijakan dihapus atau memasuki status `error` atau `disabled`, Amazon Data Lifecycle Manager berhenti menghapus snapshot. Jika Anda ingin Amazon Data Lifecycle Manager agar terus menghapus semua snapshot, termasuk yang terakhir, pilih Perpanjang penghapusan.

> **Note**
>
> Jika Anda mengaktifkan perpanjang penghapusan, Anda menimpa kedua perilaku yang dijelaskan di atas secara bersamaan.

c. Untuk menyalin snapshot yang dibuat oleh kebijakan ke Wilayah lain, pilih Buat salinan lintas-Wilayah, lalu pilih hingga 3 Wilayah tujuan.

- Jika snapshot sumber dienkripsi, atau jika enkripsi secara default diaktifkan untuk Wilayah tujuan, snapshot yang disalin dienkripsi menggunakan kunci KMS default untuk enkripsi EBS di Wilayah tujuan.
- Jika snapshot sumber tidak dienkripsi dan enkripsi secara default dinonaktifkan untuk Wilayah tujuan, snapshot yang disalin tidak dienkripsi.

10. (Opsional) Untuk menambahkan tanda ke kebijakan, pilih Tambahkan tanda lalu tentukan kunci tanda dan pasangan nilai.
11. Pilih Buat kebijakan default.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRole already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

AWS CLI

Untuk membuat kebijakan default untuk snapshot EBS

Gunakan perintah create-lifecycle-policy. Anda dapat menentukan parameter permintaan dalam salah satu dari dua metode, bergantung pada kasus penggunaan atau preferensi Anda:

- Metode 1

```
$ aws dlm create-lifecycle-policy \
--state ENABLED | DISABLED \
--description "policy_description" \
--execution-role-arn role_arn \
--default-policy VOLUME \
--create-interval creation_frequency_in_days (1-7) \
--retain-interval retention_period_in_days (2-14) \
--copy-tags | --no-copy-tags \
--extend-deletion | --no-extend-deletion \
--cross-region-copy-targets TargetRegion=destination_region_code \
--exclusions ExcludeBootVolumes=true | false,
 ExcludeTags=[{Key=tag_key,Value=tag_value}], ExcludeVolumeTypes="standard | gp2 |
 gp3 | io1 | io2 | st1 | sc1"
```

Misalnya, untuk membuat kebijakan default untuk snapshot EBS yang menargetkan semua volume di Wilayah, menggunakan peran IAM default, berjalan harian (default), dan mempertahankan snapshot selama 7 hari (default), Anda perlu menentukan parameter berikut:

```
$ aws dlm create-lifecycle-policy \
--state ENABLED \
--description "Daily default snapshot policy" \
--execution-role-arn arn:aws:iam::account_id:role/
AWSDataLifecycleManagerDefaultRole \
--default-policy VOLUME
```

- Metode 2

```
$ aws dlm create-lifecycle-policy \
--state ENABLED | DISABLED \
--description "policy_description" \
--execution-role-arn role_arn \
--default-policy VOLUME \
--policy-details file://policyDetails.json
```

Jika `policyDetails.json` mencakup berikut:

```
{
    "PolicyLanguage": "SIMPLIFIED",
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceType": "VOLUME",
    "CopyTags": true | false,
    "CreateInterval": creation_frequency_in_days (1-7),
    "RetainInterval": retention_period_in_days (2-14),
    "ExtendDeletion": true | false,
    "CrossRegionCopyTargets": [{"TargetRegion":"destination_region_code"}],
    "Exclusions": {
        "ExcludeBootVolume": true | false,
  "ExcludeVolumeTypes": ["standard | gp2 | gp3 | io1 | io2 | st1 | sc1"],
        "ExcludeTags": [{
            "Key": "exclusion_tag_key",
            "Value": "exclusion_tag_value"
        }]
    }
}
```

Buat kebijakan default untuk EBS-backed AMI

Prosedur berikut menunjukkan cara membuat kebijakan default untuk EBS-backed AMI.

Console

Untuk membuat kebijakan default untuk EBS-backed AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Lifecycle Manager lalu pilih Buat kebijakan siklus hidup.
3. Untuk Jenis kebijakan, pilih Kebijakan default dan kemudian pilih kebijakan EBS-backed AMI.
4. Untuk Deskripsi, masukkan deskripsi singkat untuk kebijakan tersebut.
5. Untuk peran IAM, pilih peran IAM yang memiliki izin untuk mengelola AMI.

   Sebaiknya pilih Default untuk menggunakan peran IAM default yang disediakan oleh Amazon Data Lifecycle Manager. Namun, Anda juga dapat menggunakan peran IAM kustom yang sebelumnya Anda buat.

6. Untuk Frekuensi pembuatan, tentukan seberapa sering Anda ingin kebijakan berjalan dan membuat AMI dari instans Anda.

   Frekuensi yang Anda tentukan juga menentukan instans yang dicadangkan. Kebijakan hanya akan mencadangkan instans yang belum dicadangkan oleh cara lain dalam frekuensi yang ditentukan. Misalnya, jika Anda menentukan frekuensi pembuatan 3 hari, kebijakan hanya akan membuat AMI dari instans yang belum dicadangkan dalam 3 hari terakhir.

7. Untuk Periode retensi, tentukan berapa lama Anda ingin kebijakan mempertahankan AMI yang dibuatnya. Ketika AMI mencapai ambang batas retensi, AMI akan secara otomatis dibatalkan pendaftarannya dan snapshot yang terkait akan dihapus. Periode retensi harus lebih besar dari atau sama dengan frekuensi pembuatan.

8. (Opsional) Konfigurasikan Parameter pengecualian untuk mengecualikan instans tertentu dari cadangan terjadwal. Instans yang dikecualikan tidak akan dicadangkan saat kebijakan berjalan.

   - Untuk mengecualikan instans yang memiliki tanda tertentu, pilih Tambahkan tanda, lalu tentukan kunci dan nilai tanda. Kebijakan tidak akan membuat AMI dari instans yang memiliki tanda yang ditentukan.

9. (Opsional) Di Pengaturan lanjutan, tentukan tindakan tambahan yang harus dilakukan kebijakan.

   a. Untuk menyalin tanda yang ditetapkan dari instans sumber ke AMI, pilih Salin tanda dari instans.

   b. Dengan Perpanjang penghapusan dinonaktifkan:

      - Jika instans sumber diakhiri, Amazon Data Lifecycle Manager terus membatalkan pendaftaran AMI yang dibuat sebelumnya hingga, tetapi tidak termasuk, yang terakhir berdasarkan periode retensi. Jika Anda ingin Amazon Data Lifecycle Manager membatalkan pendaftaran semua AMI, termasuk yang terakhir, pilih Perpanjang penghapusan.
      - Jika kebijakan dihapus atau memasuki status `error` atau `disabled`, Amazon Data Lifecycle Manager berhenti membatalkan pendaftaran AMI. Jika Anda ingin Amazon Data Lifecycle Manager terus membatalkan pendaftaran semua AMI, termasuk yang terakhir, pilih Perpanjang penghapusan.

> **Note**
>
> Jika Anda mengaktifkan perpanjangan penghapusan, Anda menimpa kedua perilaku yang dijelaskan di atas secara bersamaan.

c. Untuk menyalin AMI yang dibuat oleh kebijakan ke Wilayah lain, pilih Buat salinan lintas-Wilayah, lalu pilih hingga 3 Wilayah tujuan.

- Jika AMI sumber dienkripsi, atau jika enkripsi secara default diaktifkan untuk Wilayah tujuan, AMI yang disalin dienkripsi menggunakan kunci KMS default untuk enkripsi EBS di Wilayah tujuan.
- Jika AMI sumber tidak dienkripsi dan enkripsi secara default dinonaktifkan untuk Wilayah tujuan, AMI yang disalin tidak dienkripsi.

10. (Opsional) Untuk menambahkan tanda ke kebijakan, pilih Tambahkan tanda lalu tentukan kunci tanda dan pasangan nilai.
11. Pilih Buat kebijakan default.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRoleForAMIManagement already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

AWS CLI

Untuk membuat kebijakan default untuk EBS-backed AMI

Gunakan perintah create-lifecycle-policy. Anda dapat menentukan parameter permintaan dalam salah satu dari dua metode, bergantung pada kasus penggunaan atau preferensi Anda:

- Metode 1

```
$ aws dlm create-lifecycle-policy \
--state ENABLED | DISABLED \
--description "policy_description" \
```

```
--execution-role-arn role_arn \
--default-policy INSTANCE \
--create-interval creation_frequency_in_days (1-7) \
--retain-interval retention_period_in_days (2-14) \
--copy-tags | --no-copy-tags \
--extend-deletion | --no-extend-deletion \
--cross-region-copy-targets TargetRegion=destination_region_code \
--exclusions ExcludeTags=[{Key=tag_key,Value=tag_value}]
```

Misalnya, untuk membuat kebijakan default untuk EBS-backed AMI yang menargetkan semua instans di Wilayah, menggunakan peran IAM default, berjalan setiap hari (default), dan mempertahankan AMI selama 7 hari (default), Anda perlu menentukan parameter berikut:

```
$ aws dlm create-lifecycle-policy \
--state ENABLED \
--description "Daily default AMI policy" \
--execution-role-arn arn:aws:iam::account_id:role/
AWSDataLifecycleManagerDefaultRoleForAMIManagement \
--default-policy INSTANCE
```

- Metode 2

```
$ aws dlm create-lifecycle-policy \
--state ENABLED | DISABLED \
--description "policy_description" \
--execution-role-arn role_arn \
--default-policy INSTANCE \
--policy-details file://policyDetails.json
```

Jika `policyDetails.json` mencakup berikut:

```
{
    "PolicyLanguage": "SIMPLIFIED",
    "PolicyType": "IMAGE_MANAGEMENT",
    "ResourceType": "INSTANCE",
    "CopyTags": true | false,
    "CreateInterval": creation_frequency_in_days (1-7),
    "RetainInterval": retention_period_in_days (2-14),
    "ExtendDeletion": true | false,
 "CrossRegionCopyTargets": [{"TargetRegion":"destination_region_code"}],
    "Exclusions": {
        "ExcludeTags": [{
```

```
            "Key": "exclusion_tag_key",
            "Value": "exclusion_tag_value"
        }]
    }
}
```

Aktifkan kebijakan default Data Lifecycle Manager di seluruh akun dan Wilayah

Dengan menggunakan CloudFormation StackSets, Anda dapat mengaktifkan kebijakan default Amazon Data Lifecycle Manager di beberapa akun dan Wil AWS ayah dengan satu operasi.

Anda dapat menggunakan kumpulan tumpukan untuk mengaktifkan kebijakan default dengan salah satu cara berikut:

- Di seluruh AWS organisasi — Memastikan bahwa kebijakan default diaktifkan dan dikonfigurasi secara konsisten di seluruh AWS organisasi atau unit organisasi tertentu dalam organisasi. Ini dilakukan dengan menggunakan izin yang dikelola layanan. CloudFormation StackSets membuat peran IAM yang diperlukan atas nama Anda.
- Di seluruh AWS akun tertentu — Memastikan bahwa kebijakan default diaktifkan dan dikonfigurasi secara konsisten di seluruh akun target tertentu. Ini membutuhkan izin yang dikelola sendiri. Anda membuat peran IAM yang diperlukan untuk membangun hubungan kepercayaan antara akun administrator kumpulan tumpukan dan akun target.

Untuk informasi selengkapnya, lihat Model izin untuk kumpulan tumpukan di Panduan AWS CloudFormation Pengguna.

Gunakan prosedur berikut untuk mengaktifkan kebijakan default Amazon Data Lifecycle Manager di seluruh AWS organisasi, di seluruh OU tertentu, atau di seluruh akun target tertentu.

Prasyarat

Lakukan salah satu hal berikut, tergantung pada cara Anda mengaktifkan kebijakan default:

- (Di seluruh AWS organisasi) Anda harus mengaktifkan semua fitur di organisasi Anda dan mengaktifkan akses tepercaya dengan AWS Organizations. Anda juga harus menggunakan akun manajemen organisasi atau akun administrator yang didelegasikan.

- (Di seluruh akun target tertentu) Anda harus memberikan izin yang dikelola sendiri dengan membuat peran yang diperlukan untuk membangun hubungan tepercaya antara akun administrator kumpulan tumpukan dan akun target.

Console

Untuk mengaktifkan kebijakan default di seluruh AWS organisasi atau di seluruh akun target tertentu

1. Buka CloudFormation konsol di https://console.aws.amazon.com/cloudformation.
2. Di panel navigasi, pilih StackSets, lalu pilih Buat StackSet.
3. Untuk Iz in, lakukan salah satu hal berikut, tergantung pada cara Anda mengaktifkan kebijakan default:
   - (Di seluruh AWS organisasi) Pilih Service-managed izin.
   - (Di seluruh akun target tertentu) Pilih Self-service izin. Kemudian, untuk peran admin IAM ARN, pilih peran layanan IAM yang Anda buat untuk akun administrator, dan untuk nama peran eksekusi IAM, masukkan nama peran layanan IAM yang Anda buat di akun target.
4. Untuk Siapkan template, pilih Gunakan template sampel.
5. Untuk template Sam pel, lakukan salah satu hal berikut:
   - (Kebijakan default untuk snapshot EBS) Pilih Buat kebijakan default Amazon Data Lifecycle Manager untuk Snapshots EBS.
   - (Kebijakan default untuk EBS-backed AMI) Pilih Buat kebijakan default Amazon Data Lifecycle Manager untuk EBS-backed AMI.
6. Pilih Berikutnya.
7. Untuk StackSet nama dan StackSet deskripsi, masukkan nama deskriptif dan deskripsi singkat.
8. Di bagian Parameter, konfigurasikan pengaturan kebijakan default sesuai kebutuhan.

   > **Note**
   >
   > Untuk beban kerja kritis, kami sarankan CreateInterval = 1 hari dan RetainInterval = 7 hari.

9. Pilih Berikutnya.

10. (Opsional) Untuk Tag, tentukan tag untuk membantu Anda mengidentifikasi StackSet dan menumpuk sumber daya.
11. Untuk Ek sekusi ter kelola, pilih Aktif.
12. Pilih Berikutnya.
13. Untuk Menambahkan stack ke set stack, pilih Terapkan stack baru.
14. Lakukan salah satu hal berikut, tergantung pada cara Anda mengaktifkan kebijakan default:

    - (Di seluruh AWS organisasi) Untuk target Deployment pilih salah satu opsi berikut:
      - Untuk menerapkan di seluruh AWS organisasi, pilih Ter apkan ke organisasi.
      - Untuk menerapkan ke unit organisasi tertentu (OU), pilih Terapkan ke unit organisasi, dan kemudian untuk ID OU, masukkan ID OU. Untuk menambahkan OU tambahan, pilih Tambahkan OU lain.
    - (Di seluruh akun target tertentu) Untuk Akun, lakukan salah satu hal berikut:
      - Untuk menerapkan ke akun target tertentu, pilih Terapkan tumpukan di akun, lalu untuk Nomor akun, masukkan ID akun target.
      - Untuk menerapkan ke semua akun di OU tertentu, pilih Men yebarkan tumpukan ke semua akun di unit organisasi, dan kemudian untuk Nomor organisasi, masukkan ID OU target.
15. Untuk Penyebaran otomatis, pilih Di aktifkan.
16. Untuk Per ilaku penghapusan akun, pilih Per tahankan tumpukan.
17. Untuk Tentukan wilayah, pilih Wilayah tertentu untuk mengaktifkan kebijakan default, atau pilih Tambahkan semua Wilayah untuk mengaktifkan kebijakan default di semua Wilayah.
18. Pilih Berikutnya.
19. Tinjau pengaturan kumpulan tumpukan, pilih Saya mengakui bahwa CloudFormation mungkin membuat sumber daya IAM, lalu pilih K irim.

AWS CLI

Untuk mengaktifkan kebijakan default di seluruh AWS organisasi

1. Buat set tumpukan. Gunakan perintah create-st ack-set.

   Untuk `--permission-model`, tentukan `SERVICE_MANAGED`.

   Untuk`--template-url`, tentukan salah satu URL template berikut:

- (Kebijakan default untuk EBS-backed AMI) `https://s3.amazonaws.com/cloudformation-stackset-sample-templates-us-east-1/DataLifecycleManagerAMIDefaultPolicy.yaml`
- (Kebijakan default untuk snapshot EBS) `https://s3.amazonaws.com/cloudformation-stackset-sample-templates-us-east-1/DataLifecycleManagerEBSSnapshotDefaultPolicy.yaml`

Untuk `--parameters`, tentukan pengaturan untuk kebijakan default. Untuk parameter yang didukung, deskripsi parameter, dan nilai yang valid, unduh template menggunakan URL dan kemudian lihat template menggunakan editor teks.

Untuk `--auto-deployment`, tentukan `Enabled=true,RetainStacksOnAccountRemoval=true`.

```
$ aws cloudformation create-stack-set \
--stack-set-name stackset_name \
--permission-model SERVICE_MANAGED \
--template-url template_url \
--parameters "ParameterKey=param_name_1,ParameterValue=param_value_1"
 "ParameterKey=param_name_2,ParameterValue=param_value_2" \
--auto-deployment "Enabled=true, RetainStacksOnAccountRemoval=true"
```

2. Terapkan set tumpukan. Gunakan perintah create-st ack-instance.

Untuk `--stack-set-name`, tentukan nama kumpulan tumpukan yang Anda buat pada langkah sebelumnya.

Untuk `--deployment-targets OrganizationalUnitIds`, tentukan ID OU root untuk diterapkan ke seluruh organisasi, atau ID OU untuk diterapkan ke OU tertentu di organisasi.

Untuk `--regions`, tent AWS ukan Wilayah untuk mengaktifkan kebijakan default.

```
$ aws cloudformation create-stack-instances \
--stack-set-name stackset_name \
--deployment-targets OrganizationalUnitIds='["root_ou_id"]' | '["ou_id_1",
 "ou_id_2]' \
--regions '["region_1", "region_2"]'
```

Untuk mengaktifkan kebijakan default di seluruh akun target tertentu

1. Buat set tumpukan. Gunakan perintah create-st ack-set.

   Untuk`--template-url`, tentukan salah satu URL template berikut:

   - (Kebijakan default untuk EBS-backed AMI) `https://s3.amazonaws.com/cloudformation-stackset-sample-templates-us-east-1/DataLifecycleManagerAMIDefaultPolicy.yaml`
   - (Kebijakan default untuk snapshot EBS) `https://s3.amazonaws.com/cloudformation-stackset-sample-templates-us-east-1/DataLifecycleManagerEBSSnapshotDefaultPolicy.yaml`

   Untuk`--administration-role-arn`, tentukan ARN peran layanan IAM yang sebelumnya Anda buat untuk administrator kumpulan tumpukan.

   Untuk`--execution-role-name`, tentukan nama peran layanan IAM yang Anda buat di akun target.

   Untuk`--parameters`, tentukan pengaturan untuk kebijakan default. Untuk parameter yang didukung, deskripsi parameter, dan nilai yang valid, unduh template menggunakan URL dan kemudian lihat template menggunakan editor teks.

   Untuk `--auto-deployment`, tentukan `Enabled=true,RetainStacksOnAccountRemoval=true`.

   ```
   $ aws cloudformation create-stack-set \
   --stack-set-name stackset_name \
   --template-url template_url \
   --parameters "ParameterKey=param_name_1,ParameterValue=param_value_1"
    "ParameterKey=param_name_2,ParameterValue=param_value_2" \
   --administration-role-arn administrator_role_arn \
   --execution-role-name target_account_role \
   --auto-deployment "Enabled=true, RetainStacksOnAccountRemoval=true"
   ```

2. Terapkan set tumpukan. Gunakan perintah create-st ack-instance.

   Untuk`--stack-set-name`, tentukan nama kumpulan tumpukan yang Anda buat pada langkah sebelumnya.

Untuk `--accounts`, tentukan ID AWS akun target.

Untuk `--regions`, tent AWS ukan Wilayah untuk mengaktifkan kebijakan default.

```
$ aws cloudformation create-stack-instances \
--stack-set-name stackset_name \
--accounts '["account_ID_1","account_ID_2"]' \
--regions '["region_1", "region_2"]'
```

Buat kebijakan kustom Amazon Data Lifecycle Manager untuk snapshot EBS

Prosedur berikut ini menunjukkan cara menggunakan Amazon Data Lifecycle Manager untuk mengotomatisasi siklus hidup snapshot Amazon EBS.

Topik

- Membuat kebijakan siklus hidup snapshot
- Pertimbangan untuk kebijakan siklus hidup snapshot
- Sumber daya tambahan
- Otomatiskan snapshot yang konsisten dengan aplikasi dengan Data Lifecycle Manager
- Kasus penggunaan lain untuk skrip pra dan pasca Data Lifecycle Manager
- Cara kerja skrip pra dan pasca Amazon Data Lifecycle Manager
- Identifikasi snapshot yang dibuat dengan skrip pra dan pasca Data Lifecycle Manager
- Pantau skrip pra dan pasca Amazon Data Lifecycle Manager

Membuat kebijakan siklus hidup snapshot

Gunakan salah satu prosedur berikut ini untuk membuat kebijakan siklus hidup snapshot.

Console

Untuk membuat kebijakan snapshot

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

2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager, lalu pilih Buat kebijakan siklus hidup.
3. Pada layar Pilih jenis kebijakan, pilih Kebijakan snapshot EBS, lalu pilih Berikutnya.
4. Di bagian Sumber daya target, lakukan hal berikut ini:

   a. Untuk Jenis sumber daya target, pilih jenis sumber daya untuk pencadangan. Pilih `Volume` untuk membuat snapshot volume individu atau pilih `Instance` untuk membuat snapshot multi-volume dari volume yang dilampirkan ke suatu instans.

   b. (Outpostdan pelanggan Zona Lokal saja) Tentukan lokasi sumber daya target.

      Untuk Lokasi sumber daya target, tentukan lokasi sumber daya target.

      - Untuk menargetkan sumber daya di Wilayah, pilih AWS Wilayah. Amazon Data Lifecycle Manager akan mencadangkan semua sumber daya dari jenis tertentu yang memiliki tag target yang cocok hanya di Wilayah saat ini. Snapshot dibuat di Wilayah yang sama.
      - Untuk menargetkan sumber daya di Zona Lokal, pilih AWS Zona Lokal. Amazon Data Lifecycle Manager akan mencadangkan semua sumber daya dari jenis tertentu yang memiliki tag target yang cocok di semua Zona Lokal di Wilayah saat ini saja. Snapshot dapat dibuat di Zona Lokal yang sama dengan sumber daya sumber, atau di Wilayah induknya.
      - Untuk menargetkan sumber daya anOutpost, pilih AWS Outpost. Amazon Data Lifecycle Manager akan mencadangkan semua sumber daya dari jenis tertentu yang memiliki tag target yang cocok di semua Outposts di akun Anda. Snapshot dapat dibuat Outpost sama dengan sumber daya sumber, atau di Wilayah induknya.

   c. Untuk Tanda sumber daya target, pilih tanda sumber daya yang mengidentifikasi volume atau instans yang akan dicadangkan. Hanya sumber daya yang memiliki pasangan kunci tanda dan nilai yang ditentukan yang dicadangkan oleh kebijakan.

5. Untuk Deskripsi, masukkan deskripsi singkat untuk kebijakan tersebut.
6. Untuk peran IAM, pilih peran IAM yang memiliki izin untuk mengelola snapshot dan untuk mendeskripsikan volume serta instans. Untuk menggunakan peran default yang disediakan oleh Amazon Data Lifecycle Manager, pilih Peran default. Atau, untuk menggunakan peran IAM kustom yang Anda buat sebelumnya, pilih Pilih peran lain, lalu pilih peran yang akan digunakan.

7. Untuk Tanda kebijakan, tambahkan tanda yang akan diterapkan pada kebijakan siklus hidup. Anda dapat menggunakan tanda ini untuk mengidentifikasi dan mengategorikan kebijakan Anda.
8. Untuk Status kebijakan, pilih Aktifkan untuk memulai pelaksanaan kebijakan pada waktu yang dijadwalkan berikutnya, atau Nonaktifkan kebijakan untuk mencegah agar kebijakan tidak berjalan. Jika Anda tidak mengaktifkan kebijakan sekarang, kebijakan tidak akan mulai membuat snapshot sampai Anda mengaktifkannya secara manual setelah pembuatan.
9. (Kebijakan yang hanya menargetkan instans) Kecualikan volume dari set snapshot multi-volume.

   Secara default, Amazon Data Lifecycle Manager akan membuat snapshot dari semua volume yang terlampir ke instans yang ditargetkan. Namun, Anda dapat memilih untuk membuat snapshot dari subset volume yang dilampirkan. Di bagian Parameter, lakukan hal berikut ini:

   - Jika Anda tidak ingin membuat snapshot dari volume root yang dilampirkan ke instans yang ditargetkan, pilih Kecualikan volume root. Jika Anda memilih opsi ini, hanya volume data (non-root) yang dilampirkan ke instans yang ditargetkan yang akan disertakan dalam set snapshot multi-volume.
   - Jika Anda ingin membuat snapshot dari subset volume data (non-root) yang dilampirkan ke instans yang ditargetkan, pilih Kecualikan volume data tertentu, lalu tentukan tanda yang akan digunakan untuk mengidentifikasi volume data yang tidak boleh dibuat snapshot. Amazon Data Lifecycle Manager tidak akan membuat snapshot volume data yang memiliki tanda yang ditentukan. Amazon Data Lifecycle Manager hanya akan membuat snapshot dari volume data yang tidak memiliki tanda yang ditentukan.
10. Pilih Berikutnya.
11. Pada layar Konfigurasikan jadwal, konfigurasikan jadwal kebijakan. Kebijakan dapat memiliki hingga 4 jadwal. Jadwal 1 bersifat wajib. Jadwal 2, 3, dan 4 bersifat opsional. Untuk setiap jadwal kebijakan yang Anda tambahkan, lakukan hal berikut:

    a. Dalam bagian Detail jadwal, lakukan hal berikut:

        i. Untuk Nama jadwal, tentukan nama deskriptif untuk jadwal.

        ii. Untuk Frekuensi dan bidang terkait, konfigurasikan interval antara kebijakan yang dijalankan.

            Anda dapat mengonfigurasikan kebijakan yang berjalan sesuai jadwal harian, mingguan, bulanan, atau tahunan. Atau, pilih Ekspresi cron kustom untuk

menentukan interval hingga satu tahun. Untuk informasi selengkapnya, lihat ekspresi Cron dan nilai di Panduan EventBridge Pengguna Amazon.

> **Note**
>
> Jika Anda perlu mengaktifkan pengarsipan snapshot untuk jadwal, Anda harus memilih frekuensi bulanan atau tahunan, atau Anda harus menentukan ekspresi cron dengan frekuensi pembuatan minimal 28 hari. Jika menentukan frekuensi bulanan yang membuat snapshot pada hari tertentu dalam minggu tertentu (misalnya, Kamis kedua setiap bulan), untuk jadwal berbasis hitungan, hitungan retensi untuk tingkat arsip harus 4 atau lebih.

iii. Untuk Dimulai pada, tentukan waktu pelaksanaan kebijakan dijadwalkan untuk dimulai. Pelaksanaan kebijakan pertama dimulai dalam waktu satu jam setelah waktu yang dijadwalkan. Waktu harus dimasukkan dalam format `hh:mm` UTC.

iv. Untuk Jenis retensi, tentukan kebijakan retensi untuk snapshot yang dibuat berdasarkan jadwal.

Anda dapat mempertahankan snapshot berdasarkan total jumlah atau usianya.

- Count-based retensi
  - Dengan pengarsipan snapshot dinonaktifkan, rentangnya adalah dari 1 hingga `1000`. Saat ambang batas retensi tercapai, snapshot terlama dihapus secara permanen.
  - Dengan pengarsipan snapshot diaktifkan, rentangnya adalah dari `0` (diarsipkan segera setelah pembuatan) hingga `1000`. Saat ambang batas retensi tercapai, snapshot terlama dikonversi ke snapshot penuh dan dipindahkan ke tingkat arsip.
- Age-based retensi
  - Dengan pengarsipan snapshot dinonaktifkan, rentangnya adalah dari 1 hingga `100` tahun. Saat ambang batas retensi tercapai, snapshot terlama dihapus secara permanen.
  - Dengan pengarsipan snapshot diaktifkan, rentangnya adalah dari `0` hari (diarsipkan segera setelah pembuatan) hingga `100` tahun. Saat ambang

batas retensi tercapai, snapshot terlama dikonversi ke snapshot penuh dan dipindahkan ke tingkat arsip.

> **Note**
>
> - Semua jadwal harus memiliki jenis retensi yang sama (berbasis usia atau berbasis hitungan). Anda dapat menentukan jenis retensi hanya untuk Jadwal 1. Jadwal 2, 3, dan 4 mewarisi jenis retensi dari Jadwal 1. Setiap jadwal dapat memiliki jumlah atau periode retensi sendiri.
> - Jika Anda mengaktifkan pemulihan snapshot cepat, salinan lintas Wilayah, atau berbagi snapshot, Anda harus menentukan jumlah retensi 1 atau lebih, atau periode penyimpanan 1 hari atau lebih lama.

v. (AWS Outposts dan pelanggan Zona Lokal saja) Tentukan tujuan snapshot.

Untuk Tujuan snapshot, tentukan tujuan snapshot yang dibuat oleh kebijakan.

- Jika kebijakan menargetkan sumber daya di Wilayah, snapshot harus dibuat di Wilayah yang sama. AWS Wilayah dipilih untuk Anda.
- Jika kebijakan menargetkan sumber daya di Zona Lokal, Anda dapat membuat snapshot di Zona Lokal yang sama dengan sumber daya sumber, atau di Wilayah induknya.
- Jika kebijakan menargetkan sumber daya pada suatuOutpost, Anda dapat membuat snapshot yang Outpost sama dengan sumber daya sumber, atau di Wilayah induknya.

b. Konfigurasikan penandaan untuk snapshot.

Di bagian Penandaan, lakukan hal berikut ini:

i. Untuk menyalin semua tanda yang ditentukan pengguna dari volume sumber ke snapshot yang dibuat oleh jadwal, pilih Salin tanda dari sumber.

ii. Untuk menentukan tanda tambahan yang akan ditetapkan ke snapshot yang dibuat oleh jadwal ini, pilih Tambahkan tanda.

c. Konfigurasikan skrip pra dan pasca untuk snapshot yang konsisten dengan aplikasi.

Untuk informasi selengkapnya, lihat Otomatiskan snapshot yang konsisten dengan aplikasi dengan Data Lifecycle Manager.

d. (Kebijakan yang hanya menargetkan volume) Konfigurasikan pengarsipan snapshot.

Di bagian Pengarsipan snapshot, lakukan hal berikut:

> **Note**
>
> Anda dapat mengaktifkan pengarsipan snapshot hanya untuk satu jadwal dalam kebijakan.

i. Untuk mengaktifkan pengarsipan snapshot untuk jadwal, pilih Arsipkan snapshot yang dibuat oleh jadwal ini.

> **Note**
>
> Anda dapat mengaktifkan pengarsipan snapshot hanya jika frekuensi pembuatan snapshot bersifat bulanan atau tahunan, atau jika Anda menentukan ekspresi cron dengan frekuensi pembuatan minimal 28 hari.

ii. Tentukan aturan retensi untuk snapshot di tingkat arsip.

- Untuk jadwal berbasis jumlah, tentukan jumlah snapshot yang akan dipertahankan di tingkat arsip. Ketika ambang batas retensi tercapai, snapshot paling lama dihapus secara permanen dari tingkat arsip. Misalnya, jika Anda menentukan 3, jadwal akan mempertahankan maksimal 3 snapshot di tingkat arsip. Ketika snapshot keempat diarsipkan, yang tertua dari tiga snapshot yang ada di tingkat arsip dihapus.
- Untuk jadwal berbasis usia, tentukan periode waktu untuk mempertahankan snapshot di tingkat arsip. Ketika ambang batas retensi tercapai, snapshot paling lama dihapus secara permanen dari tingkat arsip. Misalnya, jika Anda menentukan 120 hari, jadwal akan secara otomatis menghapus snapshot dari tingkat arsip ketika mereka mencapai usia tersebut.

> **Important**
>
> Batas penyimpanan snapshot minimum adalah 90 hari. Anda harus menentukan aturan retensi yang mempertahankan snapshot setidaknya selama 90 hari.

e. Aktifkan pemulihan snapshot cepat.

Untuk mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibuat oleh jadwal, di bagian Pemulihan snapshot cepat, pilih Aktifkan pemulihan snapshot cepat. Jika Anda mengaktifkan pemulihan snapshot cepat, Anda harus memilih Zona Ketersediaan untuk tempat mengaktifkannya. Jika jadwal menggunakan jadwal retensi berbasis usia, Anda harus menentukan periode untuk mengaktifkan pemulihan snapshot cepat untuk setiap snapshot. Jika jadwal menggunakan retensi berbasis jumlah, Anda harus menentukan jumlah maksimal snapshot untuk mengaktifkan pemulihan snapshot cepat.

Jika jadwal membuat snapshot pada sebuahOutpost, Anda tidak dapat mengaktifkan pemulihan snapshot cepat. Pemulihan snapshot cepat tidak didukung dengan snapshot lokal yang disimpan di. Outpost

> **Note**
>
> Anda dikenai biaya untuk setiap menit pengaktifan pemulihan snapshot cepat untuk snapshot dalam Zona Ketersediaan tertentu. Biaya bersifat pro-rata minimal satu jam.

f. Konfigurasikan salinan lintas Wilayah.

Untuk menyalin snapshot yang dibuat oleh jadwal ke Outpost atau ke Wilayah yang berbeda, di bagian salin, pilih Aktifkan salin lintas wilayah. Cross-Region

Jika jadwal membuat snapshot di Wilayah, Anda dapat menyalin snapshot hingga tiga Wilayah tambahan atau Outposts di akun Anda. Anda harus menentukan aturan penyalinan Lintas wilayah terpisah untuk setiap Wilayah tujuan atauOutpost.

Untuk setiap Wilayah atauOutpost, Anda dapat memilih kebijakan penyimpanan yang berbeda dan Anda dapat memilih apakah akan menyalin semua tag atau tidak ada tag.

Jika snapshot sumber dienkripsi, atau jika enkripsi secara default diaktifkan, snapshot yang disalin dienkripsi. Jika snapshot sumber tidak dienkripsi, Anda dapat mengaktifkan enkripsi. Jika Anda tidak menentukan kunci KMS, snapshot dienkripsi menggunakan kunci KMS default untuk enkripsi EBS di setiap Wilayah tujuan. Jika Anda menentukan kunci KMS untuk Wilayah tujuan, peran IAM yang dipilih harus memiliki akses ke kunci KMS.

> **Note**
>
> Anda harus memastikan bahwa Anda tidak melebihi jumlah salinan snapshot bersamaan per Wilayah.

Jika kebijakan membuat snapshot pada suatuOutpost, maka Anda tidak dapat menyalin snapshot ke Wilayah atau Wilayah lain Outpost dan pengaturan Salinan Lintas wilayah tidak tersedia.

g. Konfigurasikan berbagi lintas akun.

Dalam Cross-account berbagi, konfigurasikan kebijakan untuk secara otomatis membagikan snapshot yang dibuat oleh jadwal dengan AWS akun lain. Lakukan hal-hal berikut:

i. Untuk mengaktifkan berbagi dengan AWS akun lain, pilih Aktifkan berbagi lintas akun.

ii. Untuk menambahkan akun yang dapat digunakan untuk berbagi snapshot, pilih Tambahkan akun, masukkan 12 digit ID akun AWS , dan pilih Tambah.

iii. Untuk membatalkan berbagi snapshot yang dibagikan secara otomatis setelah periode tertentu, pilih Batalkan pembagian secara otomatis. Jika Anda memilih untuk secara otomatis membatalkan pembagian snapshot yang dinagikan, periode setelah itu untuk secara otomatis membatalkan pembagian snapshot tidak dapat lebih lama dari periode untuk kebijakan mempertahankan snapshotnya. Misalnya, jika konfigurasi retensi kebijakan mempertahankan snapshot selama 5 hari, Anda dapat mengonfigurasi kebijakan untuk secara otomatis membatalkan pembagian snapshot yang dibagikan setelah periode hingga 4 hari. Hal ini berlaku untuk kebijakan dengan konfigurasi penyimpanan snapshot berbasis usia dan jumlah.

Jika Anda tidak mengaktifkan pembatalan pembagian otomatis, snapshot akan dibagikan hingga dihapus.

> **Note**
>
> Anda hanya dapat berbagi snapshot yang tidak dienkripsi atau yang dienkripsi menggunakan kunci yang dikelola pelanggan. Anda tidak dapat berbagi snapshot yang dienkripsi dengan kunci KMS enkripsi EBS default. Jika Anda berbagi snapshot terenkripsi, kemudian Anda juga harus berbagi kunci KMS yang digunakan untuk mengenkripsi volume sumber dengan akun target. Untuk informasi selengkapnya, lihat Mengizinkan pengguna di akun lain untuk menggunakan kunci KMS di Panduan Developer AWS Key Management Service .

h. Untuk menambahkan jadwal tambahan, pilih Tambahkan jadwal lain, yang terletak di bagian atas layar. Untuk setiap jadwal tambahan, lengkapi bidang seperti yang dijelaskan sebelumnya dalam topik ini.

i. Setelah Anda menambahkan jadwal yang diperlukan, pilih Tinjau kebijakan.

12. Tinjau ringkasan kebijakan, lalu pilih Buat kebijakan.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRole already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

Command line

Gunakan perintah create-lifecycle-policy untuk membuat kebijakan siklus hidup snapshot. Untuk `PolicyType`, tentukan `EBS_SNAPSHOT_MANAGEMENT`.

> **Note**
>
> Untuk menyederhanakan sintaksis, contoh berikut menggunakan file JSON, `policyDetails.json`, yang mencakup detail kebijakan.

Contoh 1—Kebijakan siklus hidup snapshot dengan dua jadwal

Contoh ini membuat kebijakan siklus hidup snapshot yang membubat snapshot dari semua volume yang memiliki kunci tanda `costcenter` dengan nilai `115`. Kebijakan tersebut mencakup dua jadwal. Jadwal pertama membuat snapshot setiap hari pada pukul 03.00 UTC. Jadwal kedua membuat snapshot mingguan setiap Jumat pukul 17.00 UTC.

```
aws dlm create-lifecycle-policy \
    --description "My volume policy" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "VOLUME"
    ],
    "TargetTags": [{
        "Key": "costcenter",
        "Value": "115"
    }],
    "Schedules": [{
        "Name": "DailySnapshots",
        "TagsToAdd": [{
            "Key": "type",
            "Value": "myDailySnapshot"
        }],
        "CreateRule": {
            "Interval": 24,
            "IntervalUnit": "HOURS",
            "Times": [
                "03:00"
```

```
            ]
        },
        "RetainRule": {
            "Count": 5
        },
        "CopyTags": false
    },
    {
        "Name": "WeeklySnapshots",
        "TagsToAdd": [{
            "Key": "type",
            "Value": "myWeeklySnapshot"
        }],
        "CreateRule": {
            "CronExpression": "cron(0 17 ? * FRI *)"
        },
        "RetainRule": {
            "Count": 5
        },
        "CopyTags": false
    }
]}
```

Jika permintaan berhasil, perintah mengembalikan ID kebijakan yang baru dibuat. Berikut ini adalah output contoh.

```
{
   "PolicyId": "policy-0123456789abcdef0"
}
```

Contoh 2—Kebijakan siklus hidup snapshot yang menargetkan instans dan membuat snapshot dari subset volume data (non-root)

Contoh ini membuat kebijakan siklus hidup snapshot yang membuat set snapshot multi-volume dari instans yang ditandai dengan `code=production`. Kebijakan ini hanya mencakup satu jadwal. Jadwal tidak membuat snapshot dari volume data yang ditandai dengan `code=temp`.

```
aws dlm create-lifecycle-policy \
    --description "My volume policy" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
```

```
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "code",
        "Value": "production"
    }],
    "Parameters": {
        "ExcludeDataVolumeTags": [{
            "Key": "code",
            "Value": "temp"
        }]
    },
    "Schedules": [{
        "Name": "DailySnapshots",
        "TagsToAdd": [{
            "Key": "type",
            "Value": "myDailySnapshot"
        }],
        "CreateRule": {
            "Interval": 24,
            "IntervalUnit": "HOURS",
            "Times": [
                "03:00"
            ]
        },
        "RetainRule": {
            "Count": 5
        },
        "CopyTags": false
    }
]}
```

Jika permintaan berhasil, perintah mengembalikan ID kebijakan yang baru dibuat. Berikut ini adalah output contoh.

```
{
```

```
    "PolicyId": "policy-0123456789abcdef0"
}
```

Contoh 3 — Kebijakan siklus hidup snapshot yang mengotomatiskan snapshot lokal Pos terdepan sumber daya

Contoh ini membuat kebijakan siklus hidup snapshot yang membuat snapshot volume yang ditandai `team=dev` di semua volume AndaOutposts. Kebijakan ini membuat snapshot Outposts sama dengan volume sumber. Kebijakan ini menciptakan snapshot setiap 12 jam mulai pukul `00:00` UTC.

```
aws dlm create-lifecycle-policy \
    --description "My local snapshot policy" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": "VOLUME",
 "ResourceLocations": "OUTPOST",
    "TargetTags": [{
        "Key": "team",
        "Value": "dev"
    }],
    "Schedules": [{
        "Name": "on-site backup",
        "CreateRule": {
            "Interval": 12,
            "IntervalUnit": "HOURS",
            "Times": [
                "00:00"
            ],
 "Location": [
  "OUTPOST_LOCAL"
 ]
        },
        "RetainRule": {
            "Count": 1
```

```
        },
        "CopyTags": false
    }
]}
```

Contoh 4 - Kebijakan siklus hidup snapshot yang membuat snapshot di Wilayah dan menyalinnya ke Pos terdepan

Kebijakan contoh berikut membuat snapshot volume yang ditandai dengan `team=dev`. Snapshot dibuat di Wilayah yang sama dengan volume sumber. Snapshot dibuat setiap 12 jam mulai pukul `00:00` UTC, dan mempertahankan maksimum 1 snapshot. Kebijakan ini juga menyalin snapshot ke Outpost`arn:aws:outposts:us-east-1:123456789012:outpost/op-1234567890abcdef0`, mengenkripsi snapshot yang disalin menggunakan kunci KMS enkripsi default, dan menyimpan salinan selama sebulan. 1

```
aws dlm create-lifecycle-policy \
    --description "Copy snapshots to Outpost" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": "VOLUME",
    "ResourceLocations": "CLOUD",
    "TargetTags": [{
        "Key": "team",
        "Value": "dev"
    }],
    "Schedules": [{
        "Name": "on-site backup",
        "CopyTags": false,
        "CreateRule": {
            "Interval": 12,
            "IntervalUnit": "HOURS",
            "Times": [
                "00:00"
            ],
            "Location": "CLOUD"
```

```
        },
        "RetainRule": {
            "Count": 1
        },
        "CrossRegionCopyRules" : [
        {
            "Target": "arn:aws:outposts:us-east-1:123456789012:outpost/
op-1234567890abcdef0",
            "Encrypted": true,
            "CopyTags": true,
            "RetainRule": {
                "Interval": 1,
                "IntervalUnit": "MONTHS"
            }
        }]
    }
]}
```

Contoh 5—Kebijakan siklus hidup snapshot dengan jadwal berbasis usia dengan pengarsipan aktif

Contoh ini membuat kebijakan siklus hidup snapshot yang menargetkan volume yang ditandai dengan `Name=Prod`. Kebijakan ini memiliki satu jadwal berbasis usia yang membuat snapshot pada hari pertama setiap bulan pada pukul 09:00. Jadwal ini mempertahankan setiap snapshot di tingkat standar selama satu hari, setelah itu memindahkannya ke tingkat arsip. Snapshot disimpan di tingkat arsip selama 90 hari sebelum dihapus.

```
aws dlm create-lifecycle-policy \
    --description "Copy snapshots to Outpost" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "ResourceTypes": [ "VOLUME"],
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "Schedules" : [
      {
        "Name": "sched1",
```

```
        "TagsToAdd": [
          {"Key":"createdby","Value":"dlm"}
        ],
        "CreateRule": {
          "CronExpression": "cron(0 9 1 * ? *)"
        },
        "CopyTags": true,
        "RetainRule":{
          "Interval": 1,
          "IntervalUnit": "DAYS"
        },
        "ArchiveRule": {
            "RetainRule":{
              "RetentionArchiveTier": {
                 "Interval": 90,
                 "IntervalUnit": "DAYS"
              }
            }
        }
      }
    ],
    "TargetTags": [
      {
        "Key": "Name",
        "Value": "Prod"
      }
    ]
}
```

Contoh 6—Kebijakan siklus hidup snapshot dengan jadwal berbasis jumlah dengan pengarsipan aktif

Contoh ini membuat kebijakan siklus hidup snapshot yang menargetkan volume yang ditandai dengan `Purpose=Test`. Kebijakan ini memiliki satu jadwal berbasis jumlah yang membuat snapshot pada hari pertama setiap bulan pada pukul 09:00. Jadwal ini mengarsipkan snapshot segera setelah pembuatan dan mempertahankan maksimal tiga snapshot di tingkat arsip.

```
aws dlm create-lifecycle-policy \
    --description "Copy snapshots to Outpost" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole \
```

```
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "ResourceTypes": [ "VOLUME"],
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "Schedules" : [
      {
        "Name": "sched1",
        "TagsToAdd": [
          {"Key":"createdby","Value":"dlm"}
        ],
        "CreateRule": {
          "CronExpression": "cron(0 9 1 * ? *)"
        },
        "CopyTags": true,
        "RetainRule":{
          "Count": 0
        },
        "ArchiveRule": {
            "RetainRule":{
              "RetentionArchiveTier": {
                 "Count": 3
              }
            }
        }
      }
    ],
    "TargetTags": [
      {
        "Key": "Purpose",
        "Value": "Test"
      }
    ]
}
```

Pertimbangan untuk kebijakan siklus hidup snapshot

Pertimbangan umum berikut ini berlaku untuk snapshot kebijakan siklus hidup:

- Kebijakan siklus hidup snapshot hanya menargetkan instans atau volume yang berada di Wilayah yang sama dengan kebijakan.
- Operasi pembuatan snapshot pertama dimulai dalam waktu satu jam setelah waktu mulai yang ditentukan. Operasi pembuatan snapshot selanjutnya dimulai dalam waktu yang dijadwalkan selama satu jam.
- Anda dapat membuat lebih dari satu kebijakan untuk mencadangkan volume atau instans. Misalnya, jika volume memiliki dua tanda, yaitu tanda A adalah target untuk kebijakan A untuk membuat snapshot setiap 12 jam, dan tanda B adalah target untuk kebijakan B untuk membuat snapshot setiap 24 jam, Amazon Data Lifecycle Manager membuat snapshot sesuai jadwal untuk kedua kebijakan. Atau, Anda dapat mencapai hasil yang sama dengan membuat satu kebijakan yang memiliki beberapa jadwal. Misalnya, Anda dapat membuat kebijakan tunggal yang hanya menargetkan tanda A, dan menentukan dua jadwal — satu untuk setiap 12 jam dan satu untuk setiap 24 jam.
- Tanda sumber daya peka huruf besar dan kecil.
- Jika Anda menghapus tanda target dari sumber daya yang ditargetkan oleh kebijakan, Amazon Data Lifecycle Manager tidak lagi mengelola snapshot yang ada di tingkat standar dan tingkat arsip; Anda harus menghapusnya secara manual jika tidak diperlukan lagi.
- Jika Anda membuat kebijakan yang menargetkan instans, dan volume baru dilampirkan ke instans target setelah kebijakan dibuat, volume yang baru ditambahkan disertakan dalam pencadangan pada saat pelaksanaan kebijakan berikutnya. Semua volume yang dilampirkan pada instans saat pelaksanaan kebijakan disertakan.
- Jika Anda membuat kebijakan dengan jadwal berbasis cron kustom yang dikonfigurasi untuk membuat hanya satu snapshot, kebijakan tidak akan secara otomatis menghapus snapshot ketika ambang retensi tercapai. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda membuat kebijakan berbasis usia dengan periode retensi lebih pendek dari frekuensi pembuatan, Amazon Data Lifecycle Manager akan selalu mempertahankan snapshot terakhir hingga snapshot berikutnya dibuat. Misalnya, jika kebijakan berbasis usia membuat satu snapshot setiap bulan dengan periode retensi tujuh hari, Amazon Data Lifecycle Manager akan mempertahankan setiap snapshot selama satu bulan, meskipun periode retensi adalah tujuh hari.

Pertimbangan berikut berlaku untuk pengarsipan snapshot:

- Anda dapat mengaktifkan pengarsipan snapshot hanya untuk kebijakan snapshot yang menargetkan volume.

- Anda dapat menentukan aturan pengarsipan hanya untuk satu jadwal untuk setiap kebijakan.
- Jika menggunakan konsol, Anda dapat mengaktifkan pengarsipan snapshot hanya jika frekuensi pembuatannya adalah bulanan atau tahunan, atau jika Anda menjadwalkan ekspresi cron dengan frekuensi pembuatan minimal 28 hari.

  Jika Anda menggunakan AWS CLI, AWS API, atau AWS SDK, Anda dapat mengaktifkan pengarsipan snapshot hanya jika jadwal memiliki ekspresi cron dengan frekuensi pembuatan setidaknya 28 hari.
- Periode retensi minimum di tingkat arsip adalah 90 hari.
- Ketika diarsipkan, snapshot dikonversi ke snapshot penuh ketika dipindahkan ke tingkat arsip. Hal ini dapat mengakibatkan biaya penyimpanan snapshot yang lebih tinggi. Untuk informasi selengkapnya, lihat Harga dan penagihan untuk pengarsipan snapshot Amazon EBS.
- Pemulihan snapshot cepat dan berbagi snapshot dinonaktifkan untuk snapshot saat diarsipkan.
- Jika, dalam kasus tahun kabisat, aturan retensi Anda menghasilkan periode penyimpanan arsip kurang dari 90 hari, Amazon Data Lifecycle Manager memastikan bahwa snapshot dipertahankan untuk periode minimum 90 hari.
- Jika Anda mengarsipkan snapshot yang dibuat oleh Amazon Data Lifecycle Manager secara manual, dan snapshot masih diarsipkan saat ambang retensi jadwal tercapai, Amazon Data Lifecycle Manager tidak lagi mengelola snapshot tersebut. Namun, jika Anda mengembalikan snapshot ke tingkat standar sebelum ambang retensi jadwal tercapai, jadwal akan terus mengelola snapshot sesuai aturan retensi.
- Jika Anda mengarsipkan snapshot yang dibuat oleh Amazon Data Lifecycle Manager secara manual, dan snapshot masih diarsipkan saat ambang penyimpanan jadwal tercapai, Amazon Data Lifecycle Manager tidak lagi mengelola snapshot tersebut. Namun, jika Anda mengarsipkan ulang snapshot sebelum ambang retensi jadwal tercapai, jadwal akan menghapus snapshot saat ambang retensi terpenuhi.
- Snapshot yang diarsipkan oleh Amazon Data Lifecycle Manager dihitung terhadap kuota `Archived snapshots per volume` dan `In-progress snapshot archives per account` Anda.
- Jika jadwal tidak dapat mengarsipkan snapshot setelah mencoba lagi selama 24 jam, snapshot tetap berada di tingkat standar dan dijadwalkan untuk dihapus berdasarkan waktu yang akan dihapus dari tingkat arsip. Misalnya, jika jadwal mengarsipkan snapshot selama 120 hari, snapshot tetap dalam tingkat standar selama 120 hari setelah pengarsipan gagal sebelum dihapus secara permanen. Untuk jadwal berbasis jumlah, snapshot tidak dihitung terhadap jumlah retensi jadwal.

- Snapshot harus diarsipkan di Wilayah yang sama dengan tempat pembuatannya. Jika Anda mengaktifkan salinan lintas Wilayah dan pengarsipan snapshot, Amazon Data Lifecycle Manager tidak mengarsipkan salinan snapshot.
- Snapshot yang diarsipkan oleh Amazon Data Lifecycle Manager ditandai dengan tanda sistem `aws:dlm:archived=true`. Selain itu, snapshot yang dibuat oleh jadwal berbasis usia yang diaktifkan arsip ditandai dengan tanda sistem `aws:dlm:expirationTime`, yang menunjukkan tanggal dan waktu snapshot dijadwalkan untuk diarsipkan.

Pertimbangan berikut berlaku untuk mengecualikan volume root dan volume data (non-root):

- Jika Anda memilih untuk mengecualikan volume boot dan menentukan tag yang akibatnya mengecualikan semua volume data tambahan yang dilampirkan ke instans, maka Amazon Data Lifecycle Manager tidak akan membuat snapshot apa pun untuk instans yang terpengaruh, dan itu akan memancarkan `SnapshotsCreateFailed` CloudWatch metrik. Untuk informasi selengkapnya, lihat Memantau kebijakan menggunakan CloudWatch.

Pertimbangan berikut berlaku untuk menghapus volume atau mengakhiri instans yang ditargetkan oleh kebijakan siklus hidup snapshot:

- Jika Anda menghapus volume atau mengakkhiri instans yang ditargetkan oleh kebijakan dengan jadwal retensi berbasis jumlah, Amazon Data Lifecycle Manager tidak lagi mengelola snapshot di tingkat standar dan tingkat arsip yang dibuat dari volume atau instans yang dihapus. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda menghapus volume atau mengakhiri instans yang ditargetkan oleh kebijakan dengan jadwal penyimpanan berbasis usia, kebijakan tersebut terus menghapus snapshot dari tingkat standar dan tingkat arsip yang dibuat dari volume atau instans yang dihapus pada jadwal yang ditentukan hingga, tetapi tidak termasuk snapshot terakhir. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.

Pertimbangan berikut ini berlaku untuk kebijakan siklus hidup snapshot dan pemulihan snapshot cepat:

- Amazon Data Lifecycle Manager dapat mengaktifkan pemulihan snapshot cepat hanya untuk snapshot dengan ukuran 16 TiB atau kurang. Untuk informasi selengkapnya, lihat Pemulihan snapshot cepat Amazon EBS.

- Snapshot yang diaktifkan untuk pemulihan snapshot cepat tetap aktif meskipun Anda menghapus atau menonaktifkan kebijakan, menonaktifkan pemulihan snapshot cepat untuk kebijakan, atau menonaktifkan pemulihan snapshot cepat untuk Zona Ketersediaan. Anda harus menonaktifkan pemulihan snapshot cepat untuk snapshot ini secara manual.
- Jika Anda mengaktifkan pemulihan snapshot cepat untuk suatu kebijakan dan melebihi jumlah maksimum snapshot yang dapat diaktifkan untuk pemulihan snapshot cepat, Amazon Data Lifecycle Manager membuat snapshot sesuai jadwal, tetapi tidak mengaktifkannya untuk pemulihan snapshot cepat. Setelah snapshot yang diaktifkan untuk pemulihan snapshot cepat dihapus, snapshot berikutnya yang dibuat Amazon Data Lifecycle Manager diaktifkan untuk pemulihan snapshot cepat.
- Ketika pemulihan snapshot cepat diaktifkan untuk snapshot, hal ini memakan waktu 60 menit per TiB untuk mengoptimalkan snapshot. Sebaiknya konfigurasikan jadwal Anda sehingga setiap snapshot sepenuhnya dioptimalkan sebelum Amazon Data Lifecycle Manager membuat snapshot berikutnya.
- Jika Anda mengaktifkan pemulihan snapshot cepat untuk kebijakan yang menargetkan instans, Amazon Data Lifecycle Manager mengaktifkan pemulihan snapshot cepat untuk setiap snapshot dalam snapshot multi-volume yang diatur secara individu. Jika gagal mengaktifkan pemulihan snapshot cepat untuk salah satu snapshot dalam set snapshot multi-volume, Amazon Data Lifecycle Manager masih akan mencoba mengaktifkan pemulihan snapshot cepat untuk snapshot yang tersisa dalam set snapshot.
- Anda dikenai biaya untuk setiap menit pengaktifan pemulihan snapshot cepat untuk snapshot dalam Zona Ketersediaan tertentu. Biaya bersifat pro-rata minimal satu jam. Untuk informasi selengkapnya, lihat Harga dan Penagihan.

> **Note**
>
> Bergantung pada konfigurasi kebijakan siklus hidup, Anda dapat mengaktifkan banyak snapshot untuk pemulihan snapshot cepat di banyak Zona Ketersediaan secara bersamaan.

Pertimbangan berikut berlaku untuk kebijakan siklus hidup snapshot dan volume Multi-Attach yang diaktifkan:

- Saat membuat kebijakan siklus hidup yang menargetkan instans yang memiliki volume aktif Multi-Attach yang sama, Amazon Data Lifecycle Manager memulai snapshot volume untuk setiap instans

terlampir. Gunakan tanda stempel waktu untuk mengidentifikasi sejumlah snapshot yang konsisten dengan waktu yang dibuat dari instans yang dilampirkan.

Pertimbangan berikut berlaku untuk berbagi snapshot antar-akun:

- Anda hanya dapat berbagi snapshot yang tidak dienkripsi atau yang dienkripsi menggunakan kunci yang dikelola pelanggan.
- Anda tidak dapat berbagi snapshot yang dienkripsi dengan kunci KMS enkripsi EBS default.
- Jika Anda berbagi snapshot terenkripsi, Anda juga harus berbagi kunci KMS yang digunakan untuk mengenkripsi volume sumber dengan akun target. Untuk informasi selengkapnya, lihat Mengizinkan pengguna di akun lain untuk menggunakan kunci KMS di Panduan Developer AWS Key Management Service .

Pertimbangan berikut berlaku untuk kebijakan snapshot dan pengarsipan snapshot:

- Jika Anda mengarsipkan snapshot yang dibuat oleh kebijakan secara manual, dan snapshot tersebut ada di tingkat arsip saat ambang penyimpanan kebijakan tercapai, Amazon Data Lifecycle Manager tidak akan menghapus snapshot tersebut. Amazon Data Lifecycle Manager tidak mengelola snapshot saat disimpan di tingkat arsip. Jika Anda tidak lagi membutuhkan snapshot yang disimpan di tingkat arsip, Anda harus menghapusnya secara manual.

Pertimbangan berikut berlaku untuk kebijakan snapshot dan Recycle Bin:

- Jika Amazon Data Lifecycle Manager menghapus snapshot dan mengirimkannya ke Keranjang Sampah saat ambang penyimpanan kebijakan tercapai, dan memulihkan snapshot dari Keranjang Sampah secara manual, Anda harus menghapus snapshot tersebut secara manual saat tidak diperlukan lagi. 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 umum berikut ini berlaku untuk kebijakan siklus hidup snapshot yang berada dalam status kesalahan:

- Untuk kebijakan dengan jadwal retensi berbasis usia, snapshot yang akan kedaluwarsa saat kebijakan berada dalam status `error` akan dipertahankan tanpa batas. Anda harus menghapus snapshot secara manual. Saat Anda mengaktifkan ulang kebijakan, Amazon Data Lifecycle Manager akan melanjutkan penghapusan snapshot karena periode retensinya kedaluwarsa.
- Untuk kebijakan dengan jadwal retensi berbasis jumlah, kebijakan berhenti membuat dan menghapus AMI saat berada dalam status `error`. Saat Anda mengaktifkan kembali kebijakan, Amazon Data Lifecycle Manager akan melanjutkan pembuatan snapshot, dan melanjutkan penghapusan snapshot saat ambang retensi terpenuhi.

Pertimbangan berikut berlaku untuk kebijakan snapshot dan kunci snapshot:

- Jika Anda mengunci snapshot yang dibuat secara manual oleh Amazon Data Lifecycle Manager, dan snapshot tersebut masih terkunci ketika ambang batas retensinya tercapai, Amazon Data Lifecycle Manager tidak lagi mengelola snapshot tersebut. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda mengunci snapshot secara manual yang dibuat dan diaktifkan untuk pemulihan snapshot cepat oleh Amazon Data Lifecycle Manager, dan snapshot masih terkunci saat ambang batas retensinya tercapai, Amazon Data Lifecycle Manager tidak akan menonaktifkan pemulihan snapshot cepat atau menghapus snapshot. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda mendaftarkan snapshot yang dibuat secara manual oleh Amazon Data Lifecycle Manager dengan AMI, lalu mengunci snapshot tersebut, dan snapshot tersebut masih terkunci serta dikaitkan dengan AMI saat ambang batas retensinya tercapai, Amazon Data Lifecycle Manager akan terus berusaha menghapus snapshot tersebut. Ketika AMI dibatalkan pendaftarannya dan snapshot tidak terkunci, Amazon Data Lifecycle Manager akan secara otomatis menghapus snapshot tersebut.

Sumber daya tambahan

Untuk informasi selengkapnya, lihat blog Men gotomatiskan snapshot Amazon EBS dan manajemen AMI menggunakan blog AWS penyimpanan Amazon Data Lifecycle Manager.

Otomatiskan snapshot yang konsisten dengan aplikasi dengan Data Lifecycle Manager

Anda dapat mengotomatiskan snapshot yang konsisten dengan aplikasi menggunakan Amazon Data Lifecycle Manager dengan mengaktifkan skrip pra dan pasca dalam kebijakan siklus hidup snapshot yang menargetkan instans.

Amazon Data Lifecycle Manager terintegrasi dengan AWS Systems Manager (Manajer Sistem) untuk mendukung snapshot yang konsisten dengan aplikasi. Amazon Data Lifecycle Manager menggunakan dokumen perintah Systems Manager (SSM) yang menyertakan skrip pra dan pasca untuk mengotomatisasi tindakan yang diperlukan untuk menyelesaikan snapshot yang konsisten dengan aplikasi. Sebelum Amazon Data Lifecycle Manager memulai pembuatan snapshot, Amazon menjalankan perintah di skrip pra untuk membekukan dan menghapus. I/O Setelah Amazon Data Lifecycle Manager memulai pembuatan snapshot, ia menjalankan perintah dalam skrip posting untuk mencairkan. I/O

Menggunakan Amazon Data Lifecycle Manager, Anda dapat mengotomatisasi snapshot yang konsisten dengan aplikasi berikut ini:

- Aplikasi Windows yang menggunakan Volume Shadow Copy Service (VSS)
- SAP HANA menggunakan dokumen SS AWS DM terkelola. Untuk informasi selengkapnya, lihat Snapshot Amazon EBS untuk SAP HANA.
- Self-managed database, seperti MySQL, PostgreSQL atau InterSystems IRIS, menggunakan template dokumen SSM

Topik

- Persyaratan untuk menggunakan skrip pra dan pasca
- Memulai snapshot yang konsisten dengan aplikasi
- Pertimbangan untuk Pencadangan VSS dengan Amazon Data Lifecycle Manager
- Tanggung jawab bersama untuk snapshot yang konsisten dengan aplikasi

Persyaratan untuk menggunakan skrip pra dan pasca

Tabel berikut menguraikan persyaratan untuk menggunakan skrip pra dan pasca dengan Amazon Data Lifecycle Manager.

	Application-consistent foto-foto		
Persyaratan	Cadangan VSS	Dokumen SSM Kustom	Kasus penggunaan lainnya
Agen SSM diinstal dan berjalan pada instans target	✓	✓	✓
Persyaratan sistem VSS terpenuhi pada instans target	✓		
Profil instance yang diaktifkan VSS terkait dengan instans target	✓		
Komponen VSS diinstal pada instans target	✓		
Siapkan dokumen SSM dengan perintah skrip pra dan pasca		✓	✓
Siapkan peran IAM Amazon Data Lifecycle Manager jalankan pra dan pasca skrip	✓	✓	✓
Buat kebijakan snapshot yang menargetkan instance	✓	✓	✓

	Application-consistent foto-foto
dan dikonfigurasi untuk skrip pra dan pasca	

Memulai snapshot yang konsisten dengan aplikasi

Bagian ini menjelaskan langkah-langkah yang perlu Anda ikuti untuk mengotomatisasi snapshot yang konsisten dengan aplikasi menggunakan Amazon Data Lifecycle Manager.

Langkah 1: Menyiapkan instans target

Anda perlu menyiapkan instans yang ditargetkan untuk snapshot yang konsisten dengan aplikasi menggunakan Amazon Data Lifecycle Manager. Lakukan salah satu langkah berikut sesuai dengan kasus penggunaan Anda.

Prepare for VSS Backups

Untuk mempersiapkan instans target Anda untuk cadangan VSS

1. Instal SSM Agent pada instans target Anda, jika belum diinstal. Jika SSM Agent sudah diinstal pada instans target Anda, lewati langkah ini.

   Untuk informasi selengkapnya, lihat Bekerja dengan Agen SSM pada instans EC2 untuk server Windows.
2. Pastikan bahwa Agen SSM sedang berjalan. Untuk informasi selengkapnya, lihat Memeriksa status SSM Agent dan memulai agen.
3. Siapkan Systems Manager untuk instans Amazon EC2. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager untuk instans Amazon EC2 di Panduan Pengguna AWS Systems Manager .
4. Pastikan persyaratan sistem untuk backup VSS terpenuhi.
5. Lampirkan profil VSS-enabled instance ke instance target.
6. Instal komponen VSS.

Prepare for SAP HANA backups

Untuk mempersiapkan instans target Anda untuk cadangan SAP HANA

1. Siapkan lingkungan SAP HANA pada instans target Anda.

   a. Siapkan instans Anda dengan SAP HANA. Jika Anda belum memiliki lingkungan SAP HANA yang ada, Anda dapat merujuk ke Penyiapan Lingkungan SAP HANA di AWS.

   b. Masuk ke SystemDB sebagai pengguna administrator yang sesuai.

   c. Buat pengguna cadangan basis data untuk digunakan dengan Amazon Data Lifecycle Manager.

   ```
   CREATE USER username PASSWORD password NO FORCE_FIRST_PASSWORD_CHANGE;
   ```

   Misalnya, perintah berikut membuat pengguna bernama `dlm_user` dengan kata sandi `password`.

   ```
   CREATE USER dlm_user PASSWORD password NO FORCE_FIRST_PASSWORD_CHANGE;
   ```

   d. Tetapkan `BACKUP OPERATOR` peran ke pengguna cadangan basis data yang Anda buat di langkah sebelumnya.

   ```
   GRANT BACKUP OPERATOR TO username
   ```

   Misalnya, perintah berikut menetapkan peran untuk pengguna bernama `dlm_user`.

   ```
   GRANT BACKUP OPERATOR TO dlm_user
   ```

   e. Masuk ke sistem operasi sebagai administrator, misalnya *sid*`adm`.

   f. Buat entri `hdbuserstore` untuk menyimpan informasi koneksi sehingga dokumen SSM SAP HANA dapat terhubung ke SAP HANA tanpa pengguna harus memasukkan informasi tersebut.

   ```
   hdbuserstore set DLM_HANADB_SNAPSHOT_USER
    localhost:3hana_instance_number13 username password
   ```

   Misalnya:

```
hdbuserstore set DLM_HANADB_SNAPSHOT_USER localhost:30013 dlm_user password
```

g. Uji koneksi.

```
hdbsql -U DLM_HANADB_SNAPSHOT_USER "select * from dummy"
```

2. Instal SSM Agent pada instans target Anda, jika belum diinstal. Jika SSM Agent sudah diinstal pada instans target Anda, lewati langkah ini.

   Untuk informasi selengkapnya, lihat Meng instal Agen SSM secara manual pada instans EC2 untuk Linux.

3. Pastikan bahwa Agen SSM sedang berjalan. Untuk informasi selengkapnya, lihat Memeriksa status SSM Agent dan memulai agen.
4. Siapkan Systems Manager untuk instans Amazon EC2. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager untuk instans Amazon EC2 di Panduan Pengguna AWS Systems Manager .

Prepare for custom SSM documents

Untuk menyiapkan dokumen SSM kustom instans target Anda

1. Instal SSM Agent pada instans target Anda, jika belum diinstal. Jika SSM Agent sudah diinstal pada instans target Anda, lewati langkah ini.
   - (Instans Linux) Meng instal Agen SSM secara manual pada instans EC2 untuk Linux
   - (Instans Windows) Bek erja dengan Agen SSM pada instans EC2 untuk Windows Server
2. Pastikan bahwa Agen SSM sedang berjalan. Untuk informasi selengkapnya, lihat Memeriksa status SSM Agent dan memulai agen.
3. Siapkan Systems Manager untuk instans Amazon EC2. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager untuk instans Amazon EC2 di Panduan Pengguna AWS Systems Manager .

Langkah 2: Siapkan Dokumen SSM

> **Note**
>
> Langkah ini hanya diperlukan untuk dokumen SSM kustom. Hal ini tidak diperlukan untuk Cadangan VSS atau SAP HANA. Untuk VSS Backup dan SAP HANA, Amazon Data Lifecycle Manager menggunakan dokumen SSM AWS terkelola.

Jika Anda mengotomatiskan snapshot yang konsisten dengan aplikasi untuk database yang dikelola sendiri, seperti MySQL, PostgreSQL, atau InterSystems IRIS, Anda harus membuat dokumen perintah SSM yang menyertakan skrip pra untuk membekukan dan menghapus I/O sebelum pembuatan snapshot dimulai, dan skrip pos untuk mencair setelah pembuatan snapshot dimulai. I/O

Jika database MySQL, PostgreSQL, atau InterSystems IRIS Anda menggunakan konfigurasi standar, Anda dapat membuat dokumen perintah SSM menggunakan contoh konten dokumen SSM di bawah ini. Jika database MySQL, PostgreSQL, atau InterSystems IRIS Anda menggunakan konfigurasi non-standar, Anda dapat menggunakan konten sampel di bawah ini sebagai titik awal untuk dokumen perintah SSM Anda dan kemudian menyesuaikannya untuk memenuhi kebutuhan Anda. Atau, jika Anda ingin membuat dokumen SSM baru dari awal, Anda dapat menggunakan templat dokumen SSM kosong di bawah ini dan menambahkan praperintah dan pascaperintah Anda di bagian dokumen yang sesuai.

> **Perhatikan hal-hal berikut:**
>
> - Adalah tanggung jawab Anda untuk memastikan bahwa dokumen SSM melakukan tindakan yang benar dan diperlukan untuk konfigurasi database Anda.
> - Snapshot dijamin konsisten dengan aplikasi hanya jika skrip pra dan pasca dalam dokumen SSM Anda berhasil membekukan, menyiram, dan mencair. I/O
> - Dokumen SSM harus menyertakan bidang wajib untuk `allowedValues`, termasuk, `pre-script`, `post-script`, dan `dry-run`. Amazon Data Lifecycle Manager akan menjalankan perintah pada instans Anda berdasarkan konten bagian tersebut. Jika dokumen SSM Anda tidak memiliki bagian tersebut, Amazon Data Lifecycle Manager akan memperlakukannya sebagai eksekusi yang gagal.

MySQL sample document content

```
###############################################################################
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

# Permission is hereby granted, free of charge, to any person obtaining a copy of
 this
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
###############################################################################
schemaVersion: '2.2'
description: Amazon Data Lifecycle Manager Pre/Post script for MySQL databases
parameters:
  executionId:
    type: String
    default: None
    description: (Required) Specifies the unique identifier associated with a pre
 and/or post execution
    allowedPattern: ^(None|[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{4}-[a-fA-F0-9]{12})$
  command:
  # Data Lifecycle Manager will trigger the pre-script and post-script actions
 during policy execution.
  # 'dry-run' option is intended for validating the document execution without
 triggering any commands
  # on the instance. The following allowedValues will allow Data Lifecycle Manager
 to successfully
  # trigger pre and post script actions.
    type: String
    default: 'dry-run'
    description: (Required) Specifies whether pre-script and/or post-script should
 be executed.
    allowedValues:
```

```
    - pre-script
    - post-script
    - dry-run

mainSteps:
- action: aws:runShellScript
  description: Run MySQL Database freeze/thaw commands
  name: run_pre_post_scripts
  precondition:
    StringEquals:
    - platformType
    - Linux
  inputs:
    runCommand:
    - |
      #!/bin/bash


 ###============================================================================###
      ### Error Codes

 ###============================================================================###
      # The following Error codes will inform Data Lifecycle Manager of the type of
 error
      # and help guide handling of the error.
      # The Error code will also be emitted via AWS Eventbridge events in the
 'cause' field.
      # 1 Pre-script failed during execution - 201
      # 2 Post-script failed during execution - 202
      # 3 Auto thaw occurred before post-script was initiated - 203
      # 4 Pre-script initiated while post-script was expected - 204
      # 5 Post-script initiated while pre-script was expected - 205
      # 6 Application not ready for pre or post-script initiation - 206

      ###====================================================================###
      ### Global variables
      ###====================================================================###
      START=$(date +%s)
      # For testing this script locally, replace the below with OPERATION=$1.
      OPERATION={{ command }}
      FS_ALREADY_FROZEN_ERROR='freeze failed: Device or resource busy'
      FS_ALREADY_THAWED_ERROR='unfreeze failed: Invalid argument'
      FS_BUSY_ERROR='mount point is busy'
```

```
    # Auto thaw is a fail safe mechanism to automatically unfreeze the application
after the
    # duration specified in the global variable below. Choose the duration based
on your
    # database application's tolerance to freeze.
    export AUTO_THAW_DURATION_SECS="60"

    # Add all pre-script actions to be performed within the function below
    execute_pre_script() {
        echo "INFO: Start execution of pre-script"
        # Check if filesystem is already frozen. No error code indicates that
filesystem
        # is not currently frozen and that the pre-script can proceed with
freezing the filesystem.
        check_fs_freeze
        # Execute the DB commands to flush the DB in preparation for snapshot
        snap_db
        # Freeze the filesystem. No error code indicates that filesystem was
succefully frozen
        freeze_fs

        echo "INFO: Schedule Auto Thaw to execute in ${AUTO_THAW_DURATION_SECS}
seconds."
        $(nohup bash -c execute_schedule_auto_thaw  >/dev/null 2>&1 &)
    }

    # Add all post-script actions to be performed within the function below
    execute_post_script() {
        echo "INFO: Start execution of post-script"
        # Unfreeze the filesystem. No error code indicates that filesystem was
successfully unfrozen.
        unfreeze_fs
        thaw_db
    }

    # Execute Auto Thaw to automatically unfreeze the application after the
duration configured
    # in the AUTO_THAW_DURATION_SECS global variable.
    execute_schedule_auto_thaw() {
        sleep ${AUTO_THAW_DURATION_SECS}
        execute_post_script
    }

    # Disable Auto Thaw if it is still enabled
```

```
     execute_disable_auto_thaw() {
         echo "INFO: Attempting to disable auto thaw if enabled"
         auto_thaw_pgid=$(pgrep -f execute_schedule_auto_thaw | xargs -i ps -hp {}
-o pgid)
         if [ -n "${auto_thaw_pgid}" ]; then
             echo "INFO: execute_schedule_auto_thaw process found with pgid
${auto_thaw_pgid}"
             sudo pkill -g ${auto_thaw_pgid}
             rc=$?
             if [ ${rc} != 0 ]; then
                 echo "ERROR: Unable to kill execute_schedule_auto_thaw process.
retval=${rc}"
             else
                 echo "INFO: Auto Thaw  has been disabled"
             fi
         fi
     }

     # Iterate over all the mountpoints and check if filesystem is already in
freeze state.
     # Return error code 204 if any of the mount points are already frozen.
     check_fs_freeze() {
         for target in $(lsblk -nlo MOUNTPOINTS)
         do
             # Freeze of the root and boot filesystems is dangerous and pre-script
does not freeze these filesystems.
             # Hence, we will skip the root and boot mountpoints while checking if
filesystem is in freeze state.
             if [ $target == '/' ]; then continue; fi
             if [[ "$target" == *"/boot"* ]]; then continue; fi

             error_message=$(sudo mount -o remount,noatime $target 2>&1)
             # Remount will be a no-op without a error message if the filesystem is
unfrozen.
             # However, if filesystem is already frozen, remount will fail with
busy error message.
             if [ $? -ne 0 ];then
                 # If the filesystem is already in frozen, return error code 204
                 if [[ "$error_message" == *"$FS_BUSY_ERROR"* ]];then
                     echo "ERROR: Filesystem ${target} already frozen. Return Error
Code: 204"
                     exit 204
                 fi
```

```
                # If the check filesystem freeze failed due to any reason other
than the filesystem already frozen, return 201
                echo "ERROR: Failed to check_fs_freeze on mountpoint $target due
to error - $errormessage"
                exit 201
            fi
        done
    }

    # Iterate over all the mountpoints and freeze the filesystem.
    freeze_fs() {
        for target in $(lsblk -nlo MOUNTPOINTS)
        do
            # Freeze of the root and boot filesystems is dangerous. Hence, skip
filesystem freeze
            # operations for root and boot mountpoints.
            if [ $target == '/' ]; then continue; fi
            if [[ "$target" == *"/boot"* ]]; then continue; fi
            echo "INFO: Freezing $target"
            error_message=$(sudo fsfreeze -f $target 2>&1)
            if [ $? -ne 0 ];then
                # If the filesystem is already in frozen, return error code 204
                if [[ "$error_message" == *"$FS_ALREADY_FROZEN_ERROR"* ]]; then
                    echo "ERROR: Filesystem ${target} already frozen. Return Error
Code: 204"
                    sudo mysql -e 'UNLOCK TABLES;'
                    exit 204
                fi
                # If the filesystem freeze failed due to any reason other than the
filesystem already frozen, return 201
                echo "ERROR: Failed to freeze mountpoint $targetdue due to error -
$errormessage"
                thaw_db
                exit 201
            fi
            echo "INFO: Freezing complete on $target"
        done
    }

    # Iterate over all the mountpoints and unfreeze the filesystem.
    unfreeze_fs() {
        for target in $(lsblk -nlo MOUNTPOINTS)
        do
```

```
            # Freeze of the root and boot filesystems is dangerous and pre-script
does not freeze these filesystems.
            # Hence, will skip the root and boot mountpoints during unfreeze as
well.
            if [ $target == '/' ]; then continue; fi
            if [[ "$target" == *"/boot"* ]]; then continue; fi
            echo "INFO: Thawing $target"
            error_message=$(sudo fsfreeze -u $target 2>&1)
            # Check if filesystem is already unfrozen (thawed). Return error code
204 if filesystem is already unfrozen.
            if [ $? -ne 0 ]; then
                if [[ "$error_message" == *"$FS_ALREADY_THAWED_ERROR"* ]]; then
                    echo "ERROR: Filesystem ${target} is already in thaw state.
Return Error Code: 205"
                    exit 205
                fi
                # If the filesystem unfreeze failed due to any reason other than
the filesystem already unfrozen, return 202
                echo "ERROR: Failed to unfreeze mountpoint $targetdue due to error
- $errormessage"
                exit 202
            fi
            echo "INFO: Thaw complete on $target"
        done
    }

    snap_db() {
        # Run the flush command only when MySQL DB service is up and running
        sudo systemctl is-active --quiet mysqld.service
        if [ $? -eq 0 ]; then
            echo "INFO: Execute MySQL Flush and Lock command."
            sudo mysql -e 'FLUSH TABLES WITH READ LOCK;'
            # If the MySQL Flush and Lock command did not succeed, return error
code 201 to indicate pre-script failure
            if [ $? -ne 0 ]; then
                echo "ERROR: MySQL FLUSH TABLES WITH READ LOCK command failed."
                exit 201
            fi
            sync
        else
            echo "INFO: MySQL service is inactive. Skipping execution of MySQL
Flush and Lock command."
        fi
    }
```

```
     thaw_db() {
         # Run the unlock command only when MySQL DB service is up and running
         sudo systemctl is-active --quiet mysqld.service
         if [ $? -eq 0 ]; then
             echo "INFO: Execute MySQL Unlock"
             sudo mysql -e 'UNLOCK TABLES;'
         else
             echo "INFO: MySQL service is inactive. Skipping execution of MySQL
 Unlock command."
         fi
     }

     export -f execute_schedule_auto_thaw
     export -f execute_post_script
     export -f unfreeze_fs
     export -f thaw_db

     # Debug logging for parameters passed to the SSM document
     echo "INFO: ${OPERATION} starting at $(date) with executionId:
 ${EXECUTION_ID}"

     # Based on the command parameter value execute the function that supports
     # pre-script/post-script operation
     case ${OPERATION} in
         pre-script)
             execute_pre_script
             ;;
         post-script)
             execute_post_script
             execute_disable_auto_thaw
             ;;
         dry-run)
             echo "INFO: dry-run option invoked - taking no action"
             ;;
         *)
             echo "ERROR: Invalid command parameter passed. Please use either pre-
script, post-script, dry-run."
             exit 1 # return failure
             ;;
     esac

     END=$(date +%s)
     # Debug Log for profiling the script time
```

```
      echo "INFO: ${OPERATION} completed at $(date). Total runtime: $((${END} -
 ${START})) seconds."
```

PostgreSQL sample document content

```
###############################################################################
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

# Permission is hereby granted, free of charge, to any person obtaining a copy of
 this
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
###############################################################################
schemaVersion: '2.2'
description: Amazon Data Lifecycle Manager Pre/Post script for PostgreSQL databases
parameters:
  executionId:
    type: String
    default: None
    description: (Required) Specifies the unique identifier associated with a pre
 and/or post execution
    allowedPattern: ^(None|[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{4}-[a-fA-F0-9]{12})$
  command:
  # Data Lifecycle Manager will trigger the pre-script and post-script actions
 during policy execution.
  # 'dry-run' option is intended for validating the document execution without
 triggering any commands
  # on the instance. The following allowedValues will allow Data Lifecycle Manager
 to successfully
  # trigger pre and post script actions.
    type: String
```

```
    default: 'dry-run'
    description: (Required) Specifies whether pre-script and/or post-script should
 be executed.
    allowedValues:
    - pre-script
    - post-script
    - dry-run

mainSteps:
- action: aws:runShellScript
  description: Run PostgreSQL Database freeze/thaw commands
  name: run_pre_post_scripts
  precondition:
    StringEquals:
    - platformType
    - Linux
  inputs:
    runCommand:
    - |
      #!/bin/bash


 ###============================================================================###
      ### Error Codes

 ###============================================================================###
      # The following Error codes will inform Data Lifecycle Manager of the type of
 error
      # and help guide handling of the error.
      # The Error code will also be emitted via AWS Eventbridge events in the
 'cause' field.
      # 1 Pre-script failed during execution - 201
      # 2 Post-script failed during execution - 202
      # 3 Auto thaw occurred before post-script was initiated - 203
      # 4 Pre-script initiated while post-script was expected - 204
      # 5 Post-script initiated while pre-script was expected - 205
      # 6 Application not ready for pre or post-script initiation - 206


 ###============================================================================###
      ### Global variables

 ###============================================================================###
      START=$(date +%s)
```

```
    OPERATION={{ command }}
    FS_ALREADY_FROZEN_ERROR='freeze failed: Device or resource busy'
    FS_ALREADY_THAWED_ERROR='unfreeze failed: Invalid argument'
    FS_BUSY_ERROR='mount point is busy'

    # Auto thaw is a fail safe mechanism to automatically unfreeze the application
after the
    # duration specified in the global variable below. Choose the duration based
on your
    # database application's tolerance to freeze.
    export AUTO_THAW_DURATION_SECS="60"

    # Add all pre-script actions to be performed within the function below
    execute_pre_script() {
        echo "INFO: Start execution of pre-script"
        # Check if filesystem is already frozen. No error code indicates that
filesystem
        # is not currently frozen and that the pre-script can proceed with
freezing the filesystem.
        check_fs_freeze
        # Execute the DB commands to flush the DB in preparation for snapshot
        snap_db
        # Freeze the filesystem. No error code indicates that filesystem was
succefully frozen
        freeze_fs

        echo "INFO: Schedule Auto Thaw to execute in ${AUTO_THAW_DURATION_SECS}
seconds."
        $(nohup bash -c execute_schedule_auto_thaw  >/dev/null 2>&1 &)
    }

    # Add all post-script actions to be performed within the function below
    execute_post_script() {
        echo "INFO: Start execution of post-script"
        # Unfreeze the filesystem. No error code indicates that filesystem was
successfully unfrozen
        unfreeze_fs
    }

    # Execute Auto Thaw to automatically unfreeze the application after the
duration configured
    # in the AUTO_THAW_DURATION_SECS global variable.
    execute_schedule_auto_thaw() {
        sleep ${AUTO_THAW_DURATION_SECS}
```

```
        execute_post_script
    }

    # Disable Auto Thaw if it is still enabled
    execute_disable_auto_thaw() {
        echo "INFO: Attempting to disable auto thaw if enabled"
        auto_thaw_pgid=$(pgrep -f execute_schedule_auto_thaw | xargs -i ps -hp {}
-o pgid)
        if [ -n "${auto_thaw_pgid}" ]; then
            echo "INFO: execute_schedule_auto_thaw process found with pgid
${auto_thaw_pgid}"
            sudo pkill -g ${auto_thaw_pgid}
            rc=$?
            if [ ${rc} != 0 ]; then
                echo "ERROR: Unable to kill execute_schedule_auto_thaw process.
retval=${rc}"
            else
                echo "INFO: Auto Thaw  has been disabled"
            fi
        fi
    }

    # Iterate over all the mountpoints and check if filesystem is already in
freeze state.
    # Return error code 204 if any of the mount points are already frozen.
    check_fs_freeze() {
        for target in $(lsblk -nlo MOUNTPOINTS)
        do
            # Freeze of the root and boot filesystems is dangerous and pre-script
does not freeze these filesystems.
            # Hence, we will skip the root and boot mountpoints while checking if
filesystem is in freeze state.
            if [ $target == '/' ]; then continue; fi
            if [[ "$target" == *"/boot"* ]]; then continue; fi

            error_message=$(sudo mount -o remount,noatime $target 2>&1)
            # Remount will be a no-op without a error message if the filesystem is
unfrozen.
            # However, if filesystem is already frozen, remount will fail with
busy error message.
            if [ $? -ne 0 ];then
                # If the filesystem is already in frozen, return error code 204
                if [[ "$error_message" == *"$FS_BUSY_ERROR"* ]];then
```

```
                    echo "ERROR: Filesystem ${target} already frozen. Return Error
Code: 204"
                    exit 204
                fi
                # If the check filesystem freeze failed due to any reason other
than the filesystem already frozen, return 201
                echo "ERROR: Failed to check_fs_freeze on mountpoint $target due
to error - $errormessage"
                exit 201
            fi
        done
    }

    # Iterate over all the mountpoints and freeze the filesystem.
    freeze_fs() {
        for target in $(lsblk -nlo MOUNTPOINTS)
        do
            # Freeze of the root and boot filesystems is dangerous. Hence, skip
filesystem freeze
            # operations for root and boot mountpoints.
            if [ $target == '/' ]; then continue; fi
            if [[ "$target" == *"/boot"* ]]; then continue; fi
            echo "INFO: Freezing $target"
            error_message=$(sudo fsfreeze -f $target 2>&1)
            if [ $? -ne 0 ];then
                # If the filesystem is already in frozen, return error code 204
                if [[ "$error_message" == *"$FS_ALREADY_FROZEN_ERROR"* ]]; then
                    echo "ERROR: Filesystem ${target} already frozen. Return Error
Code: 204"
                    exit 204
                fi
                # If the filesystem freeze failed due to any reason other than the
filesystem already frozen, return 201
                echo "ERROR: Failed to freeze mountpoint $targetdue due to error -
$errormessage"
                exit 201
            fi
            echo "INFO: Freezing complete on $target"
        done
    }

    # Iterate over all the mountpoints and unfreeze the filesystem.
    unfreeze_fs() {
        for target in $(lsblk -nlo MOUNTPOINTS)
```

```
        do
            # Freeze of the root and boot filesystems is dangerous and pre-script
does not freeze these filesystems.
            # Hence, will skip the root and boot mountpoints during unfreeze as
well.
            if [ $target == '/' ]; then continue; fi
            if [[ "$target" == *"/boot"* ]]; then continue; fi
            echo "INFO: Thawing $target"
            error_message=$(sudo fsfreeze -u $target 2>&1)
            # Check if filesystem is already unfrozen (thawed). Return error code
204 if filesystem is already unfrozen.
            if [ $? -ne 0 ]; then
                if [[ "$error_message" == *"$FS_ALREADY_THAWED_ERROR"* ]]; then
                    echo "ERROR: Filesystem ${target} is already in thaw state.
Return Error Code: 205"
                    exit 205
                fi
                # If the filesystem unfreeze failed due to any reason other than
the filesystem already unfrozen, return 202
                echo "ERROR: Failed to unfreeze mountpoint $targetdue due to error
- $errormessage"
                exit 202
            fi
            echo "INFO: Thaw complete on $target"
        done
    }

    snap_db() {
        # Run the flush command only when PostgreSQL DB service is up and running
        sudo systemctl is-active --quiet postgresql
        if [ $? -eq 0 ]; then
            echo "INFO: Execute Postgres CHECKPOINT"
            # PostgreSQL command to flush the transactions in memory to disk
            sudo -u postgres psql -c 'CHECKPOINT;'
            # If the PostgreSQL Command did not succeed, return error code 201 to
indicate pre-script failure
            if [ $? -ne 0 ]; then
                echo "ERROR: Postgres CHECKPOINT command failed."
                exit 201
            fi
            sync
        else
            echo "INFO: PostgreSQL service is inactive. Skipping execution of
CHECKPOINT command."
```

```
        fi
    }

    export -f execute_schedule_auto_thaw
    export -f execute_post_script
    export -f unfreeze_fs

    # Debug logging for parameters passed to the SSM document
    echo "INFO: ${OPERATION} starting at $(date) with executionId:
 ${EXECUTION_ID}"

    # Based on the command parameter value execute the function that supports
    # pre-script/post-script operation
    case ${OPERATION} in
        pre-script)
            execute_pre_script
            ;;
        post-script)
            execute_post_script
            execute_disable_auto_thaw
            ;;
        dry-run)
            echo "INFO: dry-run option invoked - taking no action"
            ;;
        *)
            echo "ERROR: Invalid command parameter passed. Please use either pre-
script, post-script, dry-run."
            exit 1 # return failure
            ;;
    esac

    END=$(date +%s)
    # Debug Log for profiling the script time
    echo "INFO: ${OPERATION} completed at $(date). Total runtime: $((${END} -
 ${START})) seconds."
```

InterSystems IRIS sample document content

```
###############################################################################
# MIT License
#
# Copyright (c) 2024 InterSystems
#
```

```
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in all
# copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
# SOFTWARE.
###======================================================================================###
schemaVersion: '2.2'
description: SSM Document Template for Amazon Data Lifecycle Manager Pre/Post script
 feature for InterSystems IRIS.
parameters:
  executionId:
    type: String
    default: None
    description: Specifies the unique identifier associated with a pre and/or post
 execution
    allowedPattern: ^(None|[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{4}-[a-fA-F0-9]{12})$
  command:
    type: String
    # Data Lifecycle Manager will trigger the pre-script and post-script actions.
 You can also use this SSM document with 'dry-run' for manual testing purposes.
    default: 'dry-run'
    description: (Required) Specifies whether pre-script and/or post-script should
 be executed.
    #The following allowedValues will allow Data Lifecycle Manager to successfully
 trigger pre and post script actions.
    allowedValues:
    - pre-script
    - post-script
    - dry-run

mainSteps:
```

```
- action: aws:runShellScript
  description: Run InterSystems IRIS Database freeze/thaw commands
  name: run_pre_post_scripts
  precondition:
    StringEquals:
    - platformType
    - Linux
  inputs:
    runCommand:
    - |
      #!/bin/bash

 ###============================================================================###
      ### Global variables

 ###============================================================================###
      DOCKER_NAME=iris
      LOGDIR=./
      EXIT_CODE=0
      OPERATION={{ command }}
      START=$(date +%s)

      # Check if Docker is installed
      # By default if Docker is present, script assumes that InterSystems IRIS is
 running in Docker
      # Leave only the else block DOCKER_EXEC line, if you run InterSystems IRIS
 non-containerised (and Docker is present).
      # Script assumes irissys user has OS auth enabled, change the OS user or
 supply login/password depending on your configuration.
      if command -v docker &> /dev/null
      then
        DOCKER_EXEC="docker exec $DOCKER_NAME"
      else
        DOCKER_EXEC="sudo -i -u irissys"
      fi


      # Add all pre-script actions to be performed within the function below
      execute_pre_script() {
        echo "INFO: Start execution of pre-script"

        # find all iris running instances
        iris_instances=$($DOCKER_EXEC iris qall 2>/dev/null | tail -n +3 | grep
 '^up' | cut -c5-  | awk '{print $1}')
```

```
        echo "`date`: Running iris instances $iris_instances"

        # Only for running instances
        for INST in $iris_instances; do

          echo "`date`: Attempting to freeze $INST"

          # Detailed instances specific log
          LOGFILE=$LOGDIR/$INST-pre_post.log

          #check Freeze status before starting
          $DOCKER_EXEC irissession $INST -U '%SYS'
 "##Class(Backup.General).IsWDSuspendedExt()"
          freeze_status=$?
          if [ $freeze_status -eq 5 ]; then
            echo "`date`:   ERROR: $INST IS already FROZEN"
            EXIT_CODE=204
          else
            echo "`date`:   $INST is not frozen"
            # Freeze
            # Docs: https://docs.intersystems.com/irislatest/csp/documatic/
%25CSP.Documatic.cls?LIBRARY=%25SYS&CLASSNAME=Backup.General#ExternalFreeze
            $DOCKER_EXEC irissession $INST -U '%SYS'
 "##Class(Backup.General).ExternalFreeze(\"$LOGFILE\",,,,,,600,,,300)"
            status=$?

            case $status in
              5) echo "`date`:   $INST IS FROZEN"
                ;;
              3) echo "`date`:   $INST FREEZE FAILED"
                EXIT_CODE=201
                ;;
              *) echo "`date`:   ERROR: Unknown status code: $status"
                EXIT_CODE=201
                ;;
            esac
            echo "`date`:   Completed freeze of $INST"
          fi
        done
        echo "`date`: Pre freeze script finished"
      }

      # Add all post-script actions to be performed within the function below
      execute_post_script() {
```

```
        echo "INFO: Start execution of post-script"

        # find all iris running instances
        iris_instances=$($DOCKER_EXEC iris qall 2>/dev/null | tail -n +3 | grep
 '^up' | cut -c5-  | awk '{print $1}')
        echo "`date`: Running iris instances $iris_instances"

        # Only for running instances
        for INST in $iris_instances; do

          echo "`date`: Attempting to thaw $INST"

          # Detailed instances specific log
          LOGFILE=$LOGDIR/$INST-pre_post.log

          #check Freeze status befor starting
          $DOCKER_EXEC irissession $INST -U '%SYS'
 "##Class(Backup.General).IsWDSuspendedExt()"
          freeze_status=$?
          if [ $freeze_status -eq 5 ]; then
            echo "`date`:  $INST is in frozen state"
            # Thaw
            # Docs: https://docs.intersystems.com/irislatest/csp/documatic/
%25CSP.Documatic.cls?LIBRARY=%25SYS&CLASSNAME=Backup.General#ExternalFreeze
            $DOCKER_EXEC irissession $INST -U%SYS
 "##Class(Backup.General).ExternalThaw(\"$LOGFILE\")"
            status=$?

            case $status in
              5) echo "`date`:   $INST IS THAWED"
                  $DOCKER_EXEC irissession $INST -U%SYS
 "##Class(Backup.General).ExternalSetHistory(\"$LOGFILE\")"
                 ;;
              3) echo "`date`:   $INST THAW FAILED"
                  EXIT_CODE=202
                 ;;
              *) echo "`date`:   ERROR: Unknown status code: $status"
                  EXIT_CODE=202
                 ;;
            esac
            echo "`date`:   Completed thaw of $INST"
          else
            echo "`date`:   ERROR: $INST IS already THAWED"
            EXIT_CODE=205
```

```
          fi
        done
        echo "`date`: Post thaw script finished"
      }

      # Debug logging for parameters passed to the SSM document
        echo "INFO: ${OPERATION} starting at $(date) with executionId:
 ${EXECUTION_ID}"

      # Based on the command parameter value execute the function that supports
      # pre-script/post-script operation
      case ${OPERATION} in
        pre-script)
          execute_pre_script
          ;;
        post-script)
          execute_post_script
            ;;
        dry-run)
          echo "INFO: dry-run option invoked - taking no action"
          ;;
        *)
          echo "ERROR: Invalid command parameter passed. Please use either pre-
script, post-script, dry-run."
          # return failure
          EXIT_CODE=1
          ;;
      esac

      END=$(date +%s)
      # Debug Log for profiling the script time
      echo "INFO: ${OPERATION} completed at $(date). Total runtime: $((${END} -
 ${START})) seconds."
      exit $EXIT_CODE
```

Untuk informasi selengkapnya, lihat GitHub repositori.

Empty document template

```
###======================================================================###
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

# Permission is hereby granted, free of charge, to any person obtaining a copy of
 this
```

```
# software and associated documentation files (the "Software"), to deal in the
 Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
###############################################################################
schemaVersion: '2.2'
description: SSM Document Template for Amazon Data Lifecycle Manager Pre/Post script
 feature
parameters:
  executionId:
    type: String
    default: None
    description: (Required) Specifies the unique identifier associated with a pre
 and/or post execution
    allowedPattern: ^(None|[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{4}-[a-fA-F0-9]{12})$
  command:
  # Data Lifecycle Manager will trigger the pre-script and post-script actions
 during policy execution.
  # 'dry-run' option is intended for validating the document execution without
 triggering any commands
  # on the instance. The following allowedValues will allow Data Lifecycle Manager
 to successfully
  # trigger pre and post script actions.
    type: String
    default: 'dry-run'
    description: (Required) Specifies whether pre-script and/or post-script should
 be executed.
    allowedValues:
    - pre-script
    - post-script
    - dry-run

mainSteps:
- action: aws:runShellScript
```

```
 description: Run Database freeze/thaw commands
 name: run_pre_post_scripts
 precondition:
   StringEquals:
   - platformType
   - Linux
 inputs:
   runCommand:
   - |
     #!/bin/bash


###############################################################################
     ### Error Codes

###############################################################################
     # The following Error codes will inform Data Lifecycle Manager of the type of
error
     # and help guide handling of the error.
     # The Error code will also be emitted via AWS Eventbridge events in the
'cause' field.
     # 1 Pre-script failed during execution - 201
     # 2 Post-script failed during execution - 202
     # 3 Auto thaw occurred before post-script was initiated - 203
     # 4 Pre-script initiated while post-script was expected - 204
     # 5 Post-script initiated while pre-script was expected - 205
     # 6 Application not ready for pre or post-script initiation - 206


###############################################################################
     ### Global variables

###############################################################################
     START=$(date +%s)
     # For testing this script locally, replace the below with OPERATION=$1.
     OPERATION={{ command }}

     # Add all pre-script actions to be performed within the function below
     execute_pre_script() {
         echo "INFO: Start execution of pre-script"
     }

     # Add all post-script actions to be performed within the function below
     execute_post_script() {
```

```
        echo "INFO: Start execution of post-script"
    }

    # Debug logging for parameters passed to the SSM document
    echo "INFO: ${OPERATION} starting at $(date) with executionId:
 ${EXECUTION_ID}"

    # Based on the command parameter value execute the function that supports
    # pre-script/post-script operation
    case ${OPERATION} in
        pre-script)
            execute_pre_script
            ;;
        post-script)
            execute_post_script
            ;;
        dry-run)
            echo "INFO: dry-run option invoked - taking no action"
            ;;
        *)
            echo "ERROR: Invalid command parameter passed. Please use either pre-
script, post-script, dry-run."
            exit 1 # return failure
            ;;
    esac

    END=$(date +%s)
    # Debug Log for profiling the script time
    echo "INFO: ${OPERATION} completed at $(date). Total runtime: $((${END} -
 ${START})) seconds."
```

Setelah Anda memiliki konten dokumen SSM, gunakan salah satu prosedur berikut untuk membuat dokumen SSM kustom.

Console

Untuk membuat dokumen perintah SSM

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com//systems-manager/.
2. Di panel navigasi, pilih Dokumen, lalu pilih Buat dokumen, Perintah atau Sesi.
3. Untuk Nama, masukkan nama deskriptif untuk dokumen.

4. Untuk Tipe target, pilih /AWS::EC2::Instance.
5. Untuk Jenis dokumen, pilih Perintah.
6. Di bidang Konten, pilih YAML lalu tempel konten dokumen.
7. Di bagian Tanda dokumen, tambahkan tanda dengan kunci tanda `DLMScriptsAccess`, dan nilai tanda `true`.

> **Important**
>
> `DLMScriptsAccess:true`Tag diperlukan oleh kebijakan terkel AWSDataLifecycleManagerSSMFullAccess AWS ola yang digunakan di Langkah 3: Siapkan peran IAM Manajer Siklus Hidup Data Amazon. Kebijakan menggunakan kunci syarat `aws:ResourceTag` untuk membatasi akses ke dokumen SSM yang memiliki tanda ini.

8. Pilih Buat dokumen.

AWS CLI

Untuk membuat dokumen perintah SSM

Gunakan perintah create-document. Untuk `--name`, tentukan nama deskriptif untuk dokumen. Untuk `--document-type`, tentukan `Command`. Untuk `--content`, tentukan jalur ke file .yaml dengan konten dokumen SSM. Untuk `--tags`, tentukan `"Key=DLMScriptsAccess,Value=true"`.

```
$ aws ssm create-document \
--content file://path/to/file/documentContent.yaml \
--name "document_name" \
--document-type "Command" \
--document-format YAML \
--tags "Key=DLMScriptsAccess,Value=true"
```

Langkah 3: Siapkan peran IAM Amazon Data Lifecycle Manager

> **Note**
>
> Langkah ini diperlukan jika:

- Anda membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan peran IAM kustom.
- Anda menggunakan baris perintah untuk membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan default.

Jika Anda menggunakan konsol untuk membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan peran default untuk mengelola snapshot (AWSDataLifecycleManagerDefaultRole), lewati langkah ini. Dalam hal ini, AWSDataLifecycleManagerSSMFullAccess kebijakan secara otomatis dilampirkan ke peran itu.

Anda harus memastikan bahwa peran IAM yang Anda gunakan untuk kebijakan memberikan izin kepada Amazon Data Lifecycle Manager untuk melakukan tindakan SSM yang diperlukan untuk menjalankan skrip pra dan pasca pada instans yang ditargetkan oleh kebijakan.

Amazon Data Lifecycle Manager menyediakan kebijakan terkelola (AWSDataLifecycleManagerSSMFullAccess) yang mencakup izin yang diperlukan. Anda dapat melampirkan kebijakan ini ke peran IAM untuk mengelola snapshot untuk memastikan bahwa kebijakan tersebut menyertakan izin.

Important

Kebijakan AWSDataLifecycleManagerSSMFullAccess terkelola menggunakan kunci `aws:ResourceTag` kondisi untuk membatasi akses ke dokumen SSM tertentu saat menggunakan skrip pra dan pasca. Untuk mengizinkan Amazon Data Lifecycle Manager mengakses dokumen SSM, Anda harus memastikan bahwa dokumen SSM Anda diberi tag. `DLMScriptsAccess:true`

Atau, Anda dapat membuat kebijakan kustom secara manual atau menetapkan izin yang diperlukan langsung ke peran IAM yang Anda gunakan. Anda dapat menggunakan izin yang sama yang ditentukan dalam kebijakan AWSDataLifecycleManagerSSMFullAccess terkelola, namun kunci `aws:ResourceTag` kondisi bersifat opsional. Jika Anda memutuskan untuk tidak menyertakan kunci syarat itu, Anda tidak perlu menandai dokumen SSM Anda dengan `DLMScriptsAccess:true`.

Gunakan salah satu metode berikut untuk menambahkan AWSDataLifecycleManagerSSMFullAccess kebijakan ke peran IAM Anda.

Console

Untuk melampirkan kebijakan terkelola ke peran kustom

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Cari dan pilih peran kustom Anda untuk mengelola snapshot.
4. Pada tab Izin, pilih Tambahkan izin, Lampirkan kebijakan.
5. Cari dan pilih kebijakan ter AWSDataLifecycleManagerSSMFullAccess kelola, lalu pilih Tambahkan izin.

AWS CLI

Untuk melampirkan kebijakan terkelola ke peran kustom

Gunakan perintah attach-role-policy. Untuk `---role-name`, tentukan nama peran kustom Anda. Untuk `--policy-arn`, tentukan `arn:aws:iam::aws:policy/AWSDataLifecycleManagerSSMFullAccess`.

```
$ aws iam attach-role-policy \
--policy-arn arn:aws:iam::aws:policy/AWSDataLifecycleManagerSSMFullAccess \
--role-name your_role_name
```

Langkah 4: Membuat kebijakan siklus hidup snapshot

Untuk mengotomatisasi snapshot yang konsisten dengan aplikasi, Anda harus membuat kebijakan siklus hidup snapshot yang menargetkan instans, dan mengonfigurasi skrip pra dan pasca untuk kebijakan tersebut.

Console

Untuk membuat kebijakan siklus hidup snapshot

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

2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager, lalu pilih Buat kebijakan siklus hidup.
3. Pada layar Pilih jenis kebijakan, pilih Kebijakan snapshot EBS, lalu pilih Berikutnya.
4. Di bagian Sumber daya target, lakukan hal berikut ini:

   a. Untuk Jenis sumber daya target, pilih `Instance`.

   b. Untuk Tanda sumber daya target, tentukan tanda sumber daya yang mengidentifikasi instans yang akan dicadangkan. Hanya sumber daya yang memiliki tanda tertentu yang akan dicadangkan.
5. Untuk peran IAM, pilih AWSDataLifecycleManagerDefaultRole (peran default untuk mengelola snapshot), atau pilih peran khusus yang Anda buat dan siapkan untuk skrip pra dan pasca.
6. Konfigurasikan jadwal dan opsi tambahan sesuai kebutuhan. Sebaiknya jadwalkan waktu pembuatan snapshot untuk periode waktu yang sesuai dengan beban kerja Anda, seperti selama jendela pemeliharaan.

   Untuk SAP HANA, kami menyarankan Anda mengaktifkan pemulihan snapshot cepat.

   > **Note**
   >
   > Jika Anda mengaktifkan jadwal untuk Cadangan VSS, Anda tidak dapat mengaktifkan Kecualikan volume data tertentu atau Salin tanda dari sumber.
7. Di bagian Skrip pra dan pasca, pilih Aktifkan skrip pra dan pasca, lalu lakukan hal berikut, bergantung pada beban kerja Anda:

   - Untuk membuat snapshot yang konsisten dengan aplikasi dari aplikasi Windows Anda, pilih Cadangan VSS.
   - Untuk membuat snapshot yang konsisten dengan aplikasi dari beban kerja SAP HANA Anda, pilih SAP HANA.
   - Untuk membuat snapshot yang konsisten dengan aplikasi dari semua database dan beban kerja lainnya, termasuk database MySQL, PostgreSQL, atau InterSystems IRIS yang dikelola sendiri, menggunakan dokumen SSM khusus, pilih dokumen SSM khusus.

     1. Untuk Opsi otomatisasi, pilih Skrip pra dan pasca.
     2. Untuk Dokumen SSM, pilih dokumen SSM yang Anda siapkan.
8. Bergantung pada opsi yang Anda pilih, konfigurasikan opsi tambahan berikut:

- Batas waktu skrip — (Khusus dokumen SSM kustom) Periode batas waktu sebelum Amazon Data Lifecycle Manager menggagalkan upaya menjalankan skrip jika belum selesai. Jika skrip tidak selesai dalam periode batas waktu, Amazon Data Lifecycle Manager menggagalkan upaya tersebut. Periode batas waktu berlaku untuk skrip pra dan pasca secara individual. Periode batas waktu minimum dan default-nya adalah 10 detik. Dan periode batas waktu maksimumnya adalah 120 detik.
- Coba lagi skrip yang gagal — Pilih opsi ini untuk mencoba lagi skrip yang tidak selesai dalam periode batas waktu. Jika skrip pra gagal, Amazon Data Lifecycle Manager akan mencoba ulang seluruh proses pembuatan snapshot, termasuk menjalankan skrip pra dan pasca. Jika skrip pasca gagal, Amazon Data Lifecycle Manager mencoba ulang skrip pasca saja; dalam hal ini, skrip pra akan selesai dan snapshot mungkin telah dibuat.
- Default ke snapshot crash-consistent — Pilih opsi ini ke default ke snapshot crash-consistent jika skrip pra gagal dijalankan. Ini adalah perilaku pembuatan snapshot default untuk Amazon Data Lifecycle Manager jika skrip pra dan pasca tidak diaktifkan. Jika Anda mengaktifkan percobaan ulang, Amazon Data Lifecycle Manager akan default ke snapshot crash-consistent hanya setelah semua upaya percobaan ulang habis. Jika skrip pra gagal dan Anda tidak menetapkan default ke snapshot crash-consistent, Amazon Data Lifecycle Manager tidak akan membuat snapshot untuk instans selama jadwal berjalan.

  > **Note**
  >
  > Jika Anda membuat snapshot untuk SAP HANA, maka Anda mungkin ingin menonaktifkan opsi ini. Crash-consistent snapshot beban kerja SAP HANA tidak dapat dipulihkan dengan cara yang sama.

9. Pilih Buat kebijakan default.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRole already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

AWS CLI

Untuk membuat kebijakan siklus hidup snapshot

Gunakan perintah create-lifecycle-policy, dan sertakan parameter `Scripts` dalam `CreateRule`. Untuk informasi selengkapnya tentang parameter, lihat Referensi API Amazon Data Lifecycle Manager.

```
$ aws dlm create-lifecycle-policy \
--description "policy_description" \
--state ENABLED \
--execution-role-arn iam_role_arn \
--policy-details file://policyDetails.json
```

Di mana `policyDetails.json` termasuk salah satu hal berikut, tergantung pada kasus penggunaan Anda:

- Cadangan VSS

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "tag_key",
        "Value": "tag_value"
    }],
    "Schedules": [{
        "Name": "schedule_name",
        "CreateRule": {
            "CronExpression": "cron_for_creation_frequency",
            "Scripts": [{
                "ExecutionHandler":"AWS_VSS_BACKUP",
                "ExecuteOperationOnScriptFailure":true|false,
                "MaximumRetryCount":retries (0-3)
            }]
        },
        "RetainRule": {
            "Count": retention_count
        }
    }]
```

```
}
```

- Pencadangan SAP HANA

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "tag_key",
        "Value": "tag_value"
    }],
    "Schedules": [{
        "Name": "schedule_name",
        "CreateRule": {
            "CronExpression": "cron_for_creation_frequency",
            "Scripts": [{
                "Stages": ["PRE","POST"],
                "ExecutionHandlerService":"AWS_SYSTEMS_MANAGER",
                "ExecutionHandler":"AWSSystemsManagerSAP-
CreateDLMSnapshotForSAPHANA",
                "ExecuteOperationOnScriptFailure":true|false,
                "ExecutionTimeout":timeout_in_seconds (10-120),
                "MaximumRetryCount":retries (0-3)
            }]
        },
        "RetainRule": {
            "Count": retention_count
        }
    }]
}
```

- Dokumen SSM Kustom

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "tag_key",
        "Value": "tag_value"
```

```
    }],
    "Schedules": [{
        "Name": "schedule_name",
        "CreateRule": {
            "CronExpression": "cron_for_creation_frequency",
            "Scripts": [{
                "Stages": ["PRE","POST"],
                "ExecutionHandlerService":"AWS_SYSTEMS_MANAGER",
                "ExecutionHandler":"ssm_document_name|arn",
                "ExecuteOperationOnScriptFailure":true|false,
                "ExecutionTimeout":timeout_in_seconds (10-120),
                "MaximumRetryCount":retries (0-3)
            }]
        },
        "RetainRule": {
            "Count": retention_count
        }
    }]
}
```

Pertimbangan untuk Pencadangan VSS dengan Amazon Data Lifecycle Manager

Dengan Amazon Data Lifecycle Manager, Anda dapat mencadangkan dan memulihkan aplikasi Windows berkemampuan VSS (Volume Shadow Copy Service) yang berjalan di instans Amazon EC2. Jika aplikasi memiliki penulis VSS yang terdaftar dengan Windows VSS, Amazon Data Lifecycle Manager membuat snapshot yang akan bersifat konsisten aplikasi untuk aplikasi itu.

> **Note**
>
> Amazon Data Lifecycle Manager saat ini mendukung snapshot sumber daya yang konsisten aplikasi yang berjalan di Amazon EC2 saja, khusus untuk skenario pencadangan di mana data aplikasi dapat dipulihkan dengan mengganti instans yang ada dengan instans baru yang dibuat dari cadangan. Tidak semua tipe instans atau aplikasi didukung untuk pencadangan VSS. Untuk informasi selengkapnya, lihat Snapshot Application-consistent Windows VSS di Panduan Pengguna Amazon EC2.

Tipe instans yang didukung

Tipe instans Amazon EC2 berikut tidak didukung untuk pencadangan VSS. Jika kebijakan Anda menargetkan salah satu tipe instans ini, Amazon Data Lifecycle Manager mungkin masih membuat cadangan VSS, tetapi snapshot mungkin tidak ditandai dengan tanda sistem yang diperlukan. Tanpa tanda ini, snapshot tidak akan dikelola oleh Amazon Data Lifecycle Manager setelah pembuatan. Anda mungkin perlu menghapus snapshot tersebut secara manual.

- T3: | `t3.nano` `t3.micro`
- T3a: | `t3a.nano` `t3a.micro`
- T2: | `t2.nano` `t2.micro`

Tanggung jawab bersama untuk snapshot yang konsisten dengan aplikasi

Anda harus memastikan bahwa:

- SSM Agent diinstal, diperbarui, dan berjalan pada instans target Anda
- Systems Manager memiliki izin untuk melakukan tindakan yang diperlukan pada instans target
- Amazon Data Lifecycle Manager memiliki izin untuk melakukan tindakan Systems Manager yang diperlukan untuk menjalankan skrip pra dan pasca pada instans target.
- Untuk beban kerja khusus, seperti database MySQL, PostgreSQL, atau InterSystems IRIS yang dikelola sendiri, dokumen SSM yang Anda gunakan menyertakan tindakan yang benar dan diperlukan untuk pembekuan, pembilasan, dan pencairan untuk konfigurasi database Anda. I/O
- Waktu pembuatan snapshot selaras dengan jadwal beban kerja Anda. Misalnya, cobalah untuk menjadwalkan pembuatan snapshot selama jendela pemeliharaan terjadwal.

Amazon Data Lifecycle Manager memastikan bahwa:

- Pembuatan snapshot dimulai dalam waktu 60 menit dari waktu pembuatan snapshot yang dijadwalkan.
- Skrip pra dijalankan sebelum pembuatan snapshot dimulai.
- Skrip pasca berjalan setelah skrip pra berhasil dan pembuatan snapshot telah dimulai. Amazon Data Lifecycle Manager menjalankan skrip pasca hanya jika skrip pra berhasil. Jika skrip pra gagal, Amazon Data Lifecycle Manager tidak akan menjalankan skrip pasca.
- Snapshot ditandai dengan tanda yang sesuai pada pembuatan.
- CloudWatch metrik dan peristiwa dipancarkan saat skrip dimulai, dan ketika gagal atau berhasil.

Kasus penggunaan lain untuk skrip pra dan pasca Data Lifecycle Manager

Selain menggunakan skrip pra dan pasca untuk mengotomatiskan snapshot yang konsisten dengan aplikasi, Anda dapat menggunakan skrip pra dan pasca bersama-sama, atau secara individual, untuk mengotomatiskan tugas administratif lainnya sebelum atau sesudah pembuatan snapshot. Misalnya:

- Menggunakan skrip pra untuk menerapkan patch sebelum membuat snapshot. Ini dapat membantu Anda membuat snapshot setelah menerapkan pembaruan perangkat lunak mingguan atau bulanan reguler Anda.

  > **Note**
  >
  > Jika Anda memilih untuk menjalankan skrip pra saja, Tetapkan default ke snapshot crash-consistent diaktifkan secara default.

- Menggunakan skrip pasca untuk menerapkan patch sebelum membuat snapshot. Ini dapat membantu Anda membuat snapshot setelah menerapkan pembaruan perangkat lunak mingguan atau bulanan reguler Anda.

Memulai untuk kasus penggunaan lainnya

Bagian ini menjelaskan langkah-langkah yang perlu Anda lakukan saat menggunakan skrip pra and/or pasca untuk kasus penggunaan selain snapshot yang konsisten dengan aplikasi.

Langkah 1: Menyiapkan instans target

Untuk mempersiapkan instans target Anda untuk skrip pra and/or -pasca

1. Instal SSM Agent pada instans target Anda, jika belum diinstal. Jika SSM Agent sudah diinstal pada instans target Anda, lewati langkah ini.
   - (Instans Linux) Meng instal Agen SSM secara manual pada instans EC2 untuk Linux
   - (Instans Windows) Bek erja dengan Agen SSM pada instans EC2 untuk Windows Server
2. Pastikan bahwa Agen SSM sedang berjalan. Untuk informasi selengkapnya, lihat Memeriksa status SSM Agent dan memulai agen.
3. Siapkan Systems Manager untuk instans Amazon EC2. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager untuk instans Amazon EC2 di Panduan Pengguna AWS Systems Manager .

Langkah 2: Siapkan Dokumen SSM

Anda harus membuat dokumen perintah SSM yang menyertakan skrip pra and/or post dengan perintah yang ingin Anda jalankan.

Anda dapat membuat dokumen SSM menggunakan templat dokumen SSM kosong di bawah ini dan menambahkan perintah pra dan pasca Anda di bagian dokumen yang sesuai.

⚠ Perhatikan hal-hal berikut:

- Adalah tanggung jawab Anda untuk memastikan bahwa dokumen SSM melakukan tindakan yang benar dan diperlukan untuk beban kerja Anda.
- Dokumen SSM harus menyertakan bidang wajib untuk `allowedValues`, termasuk, `pre-script`, `post-script`, dan `dry-run`. Amazon Data Lifecycle Manager akan menjalankan perintah pada instans Anda berdasarkan konten bagian tersebut. Jika dokumen SSM Anda tidak memiliki bagian tersebut, Amazon Data Lifecycle Manager akan memperlakukannya sebagai eksekusi yang gagal.

```
###======================================================================###
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

# Permission is hereby granted, free of charge, to any person obtaining a copy of this
# software and associated documentation files (the "Software"), to deal in the Software
# without restriction, including without limitation the rights to use, copy, modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
###======================================================================###
schemaVersion: '2.2'
description: SSM Document Template for Amazon Data Lifecycle Manager Pre/Post script
 feature
parameters:
  executionId:
    type: String
```

```
    default: None
    description: (Required) Specifies the unique identifier associated with a pre and/
or post execution
    allowedPattern: ^(None|[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-
[a-fA-F0-9]{12})$
  command:
  # Data Lifecycle Manager will trigger the pre-script and post-script actions during
 policy execution.
  # 'dry-run' option is intended for validating the document execution without
 triggering any commands
  # on the instance. The following allowedValues will allow Data Lifecycle Manager to
 successfully
  # trigger pre and post script actions.
    type: String
    default: 'dry-run'
    description: (Required) Specifies whether pre-script and/or post-script should be
 executed.
    allowedValues:
    - pre-script
    - post-script
    - dry-run

mainSteps:
- action: aws:runShellScript
  description: Run Database freeze/thaw commands
  name: run_pre_post_scripts
  precondition:
    StringEquals:
    - platformType
    - Linux
  inputs:
    runCommand:
    - |
      #!/bin/bash


 ###========================================================================###
      ### Error Codes

 ###========================================================================###
      # The following Error codes will inform Data Lifecycle Manager of the type of
 error
      # and help guide handling of the error.
```

```
    # The Error code will also be emitted via AWS Eventbridge events in the 'cause'
field.
    # 1 Pre-script failed during execution - 201
    # 2 Post-script failed during execution - 202
    # 3 Auto thaw occurred before post-script was initiated - 203
    # 4 Pre-script initiated while post-script was expected - 204
    # 5 Post-script initiated while pre-script was expected - 205
    # 6 Application not ready for pre or post-script initiation - 206


###============================================================================###
    ### Global variables

###============================================================================###
    START=$(date +%s)
    # For testing this script locally, replace the below with OPERATION=$1.
    OPERATION={{ command }}

    # Add all pre-script actions to be performed within the function below
    execute_pre_script() {
        echo "INFO: Start execution of pre-script"
    }

    # Add all post-script actions to be performed within the function below
    execute_post_script() {
        echo "INFO: Start execution of post-script"
    }

    # Debug logging for parameters passed to the SSM document
    echo "INFO: ${OPERATION} starting at $(date) with executionId: ${EXECUTION_ID}"

    # Based on the command parameter value execute the function that supports
    # pre-script/post-script operation
    case ${OPERATION} in
        pre-script)
            execute_pre_script
            ;;
        post-script)
            execute_post_script
            ;;
        dry-run)
            echo "INFO: dry-run option invoked - taking no action"
            ;;
        *)
```

```
            echo "ERROR: Invalid command parameter passed. Please use either pre-
script, post-script, dry-run."
            exit 1 # return failure
            ;;
      esac

      END=$(date +%s)
      # Debug Log for profiling the script time
      echo "INFO: ${OPERATION} completed at $(date). Total runtime: $((${END} -
 ${START})) seconds."
```

Langkah 3: Siapkan peran IAM Amazon Data Lifecycle Manager

> **Note**
>
> Langkah ini diperlukan jika:
>
> - Anda membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan peran IAM kustom.
> - Anda menggunakan baris perintah untuk membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan default.
>
> Jika Anda menggunakan konsol untuk membuat atau memperbarui kebijakan snapshot pre/post berkemampuan skrip yang menggunakan peran default untuk mengelola snapshot (AWSDataLifecycleManagerDefaultRole), lewati langkah ini. Dalam hal ini, AWSDataLifecycleManagerSSMFullAccess kebijakan secara otomatis dilampirkan ke peran itu.

Anda harus memastikan bahwa peran IAM yang Anda gunakan untuk kebijakan memberikan izin kepada Amazon Data Lifecycle Manager untuk melakukan tindakan SSM yang diperlukan untuk menjalankan skrip pra dan pasca pada instans yang ditargetkan oleh kebijakan.

Amazon Data Lifecycle Manager menyediakan kebijakan terkelola (AWSDataLifecycleManagerSSMFullAccess) yang mencakup izin yang diperlukan. Anda dapat melampirkan kebijakan ini ke peran IAM untuk mengelola snapshot untuk memastikan bahwa kebijakan tersebut menyertakan izin.

> **Important**
>
> Kebijakan AWSDataLifecycleManagerSSMFullAccess terkelola menggunakan kunci `aws:ResourceTag` kondisi untuk membatasi akses ke dokumen SSM tertentu saat menggunakan skrip pra dan pasca. Untuk mengizinkan Amazon Data Lifecycle Manager mengakses dokumen SSM, Anda harus memastikan bahwa dokumen SSM Anda diberi tag. `DLMScriptsAccess:true`

Atau, Anda dapat membuat kebijakan kustom secara manual atau menetapkan izin yang diperlukan langsung ke peran IAM yang Anda gunakan. Anda dapat menggunakan izin yang sama yang ditentukan dalam kebijakan AWSDataLifecycleManagerSSMFullAccess terkelola, namun kunci `aws:ResourceTag` kondisi bersifat opsional. Jika Anda memutuskan untuk tidak menyertakan kunci syarat itu, Anda tidak perlu menandai dokumen SSM Anda. `DLMScriptsAccess:true`

Gunakan salah satu metode berikut untuk menambahkan AWSDataLifecycleManagerSSMFullAccess kebijakan ke peran IAM Anda.

Console

Untuk melampirkan kebijakan terkelola ke peran kustom

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Cari dan pilih peran kustom Anda untuk mengelola snapshot.
4. Pada tab Izin, pilih Tambahkan izin, Lampirkan kebijakan.
5. Cari dan pilih kebijakan ter AWSDataLifecycleManagerSSMFullAccess kelola, lalu pilih Tambahkan izin.

AWS CLI

Untuk melampirkan kebijakan terkelola ke peran kustom

Gunakan perintah attach-role-policy. Untuk `---role-name`, tentukan nama peran kustom Anda. Untuk `--policy-arn`, tentukan `arn:aws:iam::aws:policy/AWSDataLifecycleManagerSSMFullAccess`.

```
$ aws iam attach-role-policy \
```

```
--policy-arn arn:aws:iam::aws:policy/AWSDataLifecycleManagerSSMFullAccess \
--role-name your_role_name
```

Membuat kebijakan siklus hidup snapshot

Console

Untuk membuat kebijakan siklus hidup snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager, lalu pilih Buat kebijakan siklus hidup.
3. Pada layar Pilih jenis kebijakan, pilih Kebijakan snapshot EBS, lalu pilih Berikutnya.
4. Di bagian Sumber daya target, lakukan hal berikut ini:
   a. Untuk Jenis sumber daya target, pilih `Instance`.
   b. Untuk Tanda sumber daya target, tentukan tanda sumber daya yang mengidentifikasi instans yang akan dicadangkan. Hanya sumber daya yang memiliki tanda tertentu yang akan dicadangkan.
5. Untuk peran IAM, pilih AWSDataLifecycleManagerDefaultRole (peran default untuk mengelola snapshot), atau pilih peran khusus yang Anda buat dan siapkan untuk skrip pra dan pasca.
6. Konfigurasikan jadwal dan opsi tambahan sesuai kebutuhan. Sebaiknya jadwalkan waktu pembuatan snapshot untuk periode waktu yang sesuai dengan beban kerja Anda, seperti selama jendela pemeliharaan.
7. Di bagian Skrip pra dan pasca, pilih Aktifkan skrip pra dan pasca, lalu lakukan hal berikut:
   a. Pilih Dokumen SSM Kustom.
   b. Untuk Opsi otomatis, pilih opsi yang cocok dengan skrip yang ingin Anda jalankan.
   c. Untuk Dokumen SSM, pilih dokumen SSM yang Anda siapkan.
8. Konfigurasikan opsi tambahan berikut jika diperlukan:
   - Batas waktu skrip - Periode batas waktu setelah Amazon Data Lifecycle Manager menggagalkan upaya menjalankan skrip jika belum selesai. Jika skrip tidak selesai dalam periode batas waktu, Amazon Data Lifecycle Manager menggagalkan upaya tersebut. Periode batas waktu berlaku untuk skrip pra dan pasca secara individual. Periode batas

waktu minimum dan default-nya adalah 10 detik. Dan periode batas waktu maksimumnya adalah 120 detik.

- Coba lagi skrip yang gagal — Pilih opsi ini untuk mencoba lagi skrip yang tidak selesai dalam periode batas waktu. Jika skrip pra gagal, Amazon Data Lifecycle Manager akan mencoba ulang seluruh proses pembuatan snapshot, termasuk menjalankan skrip pra dan pasca. Jika skrip pasca gagal, Amazon Data Lifecycle Manager mencoba ulang skrip pasca saja; dalam hal ini, skrip pra akan selesai dan snapshot mungkin telah dibuat.
- Default ke snapshot crash-consistent — Pilih opsi ini ke default ke snapshot crash-consistent jika skrip pra gagal dijalankan. Ini adalah perilaku pembuatan snapshot default untuk Amazon Data Lifecycle Manager jika skrip pra dan pasca tidak diaktifkan. Jika Anda mengaktifkan percobaan ulang, Amazon Data Lifecycle Manager akan default ke snapshot crash-consistent hanya setelah semua upaya percobaan ulang habis. Jika skrip pra gagal dan Anda tidak menetapkan default ke snapshot crash-consistent, Amazon Data Lifecycle Manager tidak akan membuat snapshot untuk instans selama jadwal berjalan.

9. Pilih Buat kebijakan default.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRole already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

AWS CLI

Untuk membuat kebijakan siklus hidup snapshot

Gunakan perintah create-lifecycle-policy, dan sertakan parameter `Scripts` dalam `CreateRule`. Untuk informasi selengkapnya tentang parameter, lihat Referensi API Amazon Data Lifecycle Manager.

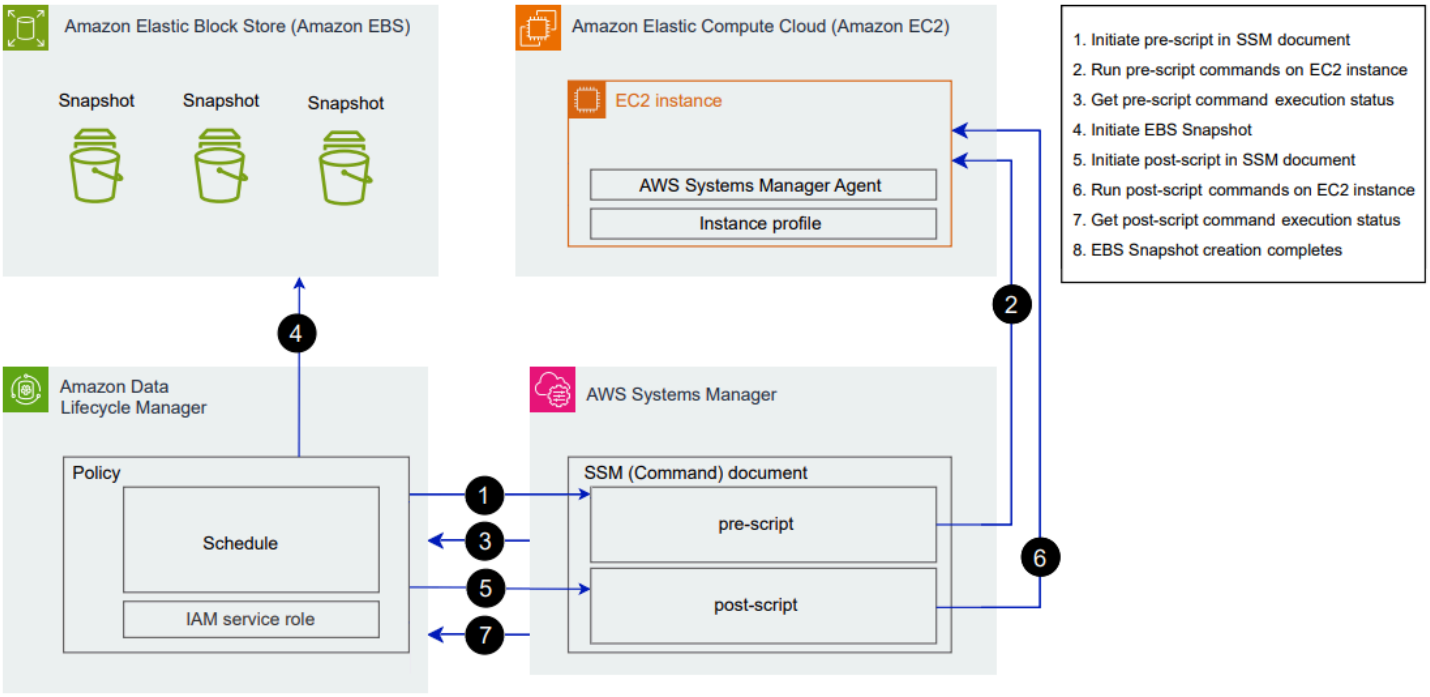
```
$ aws dlm create-lifecycle-policy \
--description "policy_description" \
--state ENABLED \
--execution-role-arn iam_role_arn \
--policy-details file://policyDetails.json
```

Di mana `policyDetails.json` termasuk yang berikut.

```
{
    "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "tag_key",
        "Value": "tag_value"
    }],
    "Schedules": [{
        "Name": "schedule_name",
        "CreateRule": {
            "CronExpression": "cron_for_creation_frequency",
            "Scripts": [{
                "Stages": ["PRE" | "POST" | "PRE","POST"],
                "ExecutionHandlerService":"AWS_SYSTEMS_MANAGER",
                "ExecutionHandler":"ssm_document_name|arn",
                "ExecuteOperationOnScriptFailure":true|false,
                "ExecutionTimeout":timeout_in_seconds (10-120),
                "MaximumRetryCount":retries (0-3)
            }]
        },
        "RetainRule": {
            "Count": retention_count
        }
    }]
}
```

Cara kerja skrip pra dan pasca Amazon Data Lifecycle Manager

Gambar berikut menunjukkan alur proses untuk skrip pra dan pasca saat menggunakan dokumen SSM kustom. Hal ini tidak berlaku untuk Pencadangan VSS.



Pada waktu pembuatan snapshot yang dijadwalkan, tindakan berikut dan interaksi lintas layanan terjadi.

1. Amazon Data Lifecycle Manager memulai tindakan skrip pra dengan memanggil dokumen SSM dan meneruskan parameter `pre-script`.

   > **Note**
   >
   > Langkah 1 hingga 3 hanya terjadi jika Anda menjalankan skrip pra. Jika Anda menjalankan skrip pasca saja, langkah 1 hingga 3 dilewati.

2. Systems Manager mengirimkan perintah pra skrip ke SSM Agent yang berjalan pada instans target. SSM Agent menjalankan perintah pada instans, dan mengirimkan informasi status kembali ke Systems Manager.

   Misalnya, jika dokumen SSM digunakan untuk membuat snapshot yang konsisten dengan aplikasi, skrip pra mungkin membeku dan dibilas I/O untuk memastikan bahwa semua data buffer ditulis ke volume sebelum snapshot diambil.

3. Systems Manager mengirimkan pembaruan status perintah skrip pra ke Amazon Data Lifecycle Manager. Jika skrip pra gagal, Amazon Data Lifecycle Manager mengambil salah satu tindakan berikut, tergantung pada cara Anda mengonfigurasi opsi skrip pra dan pasca:

Percobaan ulang	Default ke snapshot crash-consistent	Tindakan
Diaktifkan dengan percobaan ulang yang tersisa	Aktif	Coba lagi skrip sampai berhasil atau percobaan ulang habis
Habis tanpa penyelesaian yang berhasil	Aktif	Buat snapshot crash-consistent, dan jangan jalankan skrip pasca.
Diaktifkan dengan percobaan ulang yang tersisa	Nonaktif	Coba lagi skrip sampai berhasil atau percobaan ulang habis
Habis tanpa penyelesaian yang berhasil	Nonaktif	Lewati pembuatan snapshot untuk instans target, dan jangan jalankan skrip pasca.
Nonaktif	Aktif	Buat snapshot crash-consistent, dan jangan jalankan skrip pasca.
Nonaktif	Nonaktif	Lewati pembuatan snapshot untuk instans target, dan jangan jalankan skrip pasca.

4. Amazon Data Lifecycle Manager memulai pembuatan snapshot.
5. Amazon Data Lifecycle Manager memulai tindakan pasca skrip dengan memanggil dokumen SSM dan meneruskan parameter `post-script`.

> **Note**
>
> Langkah 5 hingga 7 hanya terjadi jika Anda menjalankan skrip pra. Jika Anda menjalankan skrip pasca saja, langkah 1 hingga 3 dilewati.

6. Systems Manager mengirimkan perintah post script ke SSM Agent yang berjalan pada instans target. SSM Agent menjalankan perintah pada instans, dan mengirimkan informasi status kembali ke Systems Manager.

   Misalnya, jika dokumen SSM mengaktifkan snapshot yang konsisten dengan aplikasi, skrip posting ini mungkin mencair I/O untuk memastikan bahwa database Anda melanjutkan I/O operasi normal setelah snapshot diambil.

7. Jika Anda menjalankan skrip pasca dan Systems Manager menunjukkan bahwa itu selesai dengan sukses, proses selesai.

   Jika skrip pasca gagal, Amazon Data Lifecycle Manager mengambil salah satu tindakan berikut, tergantung pada cara Anda mengonfigurasi opsi skrip pra dan pasca:

Percobaan ulang	Tindakan
Diaktifkan dengan percobaan ulang yang tersisa	Coba lagi skrip sampai berhasil atau percobaan ulang habis
Lelah tanpa sukses	Lewati skrip pasca
Nonaktif	Lewati skrip pasca

   Perlu diingat bahwa jika skrip pasca gagal, skrip pra (jika diaktifkan) akan berhasil diselesaikan, dan snapshot mungkin telah dibuat. Anda mungkin perlu mengambil tindakan lebih lanjut pada instance untuk memastikan bahwa instans beroperasi seperti yang diharapkan. Misalnya jika skrip pra dijeda dan dibuang I/O, tetapi skrip pos gagal mencair I/O, Anda mungkin perlu mengonfigurasi database Anda untuk mencair secara otomatis I/O atau Anda perlu mencairkan secara manual. I/O

8. Proses pembuatan snapshot mungkin selesai setelah skrip pasca selesai. Waktu yang dibutuhkan untuk menyelesaikan snapshot tergantung pada ukuran snapshot.

Identifikasi snapshot yang dibuat dengan skrip pra dan pasca Data Lifecycle Manager

Amazon Data Lifecycle Manager secara otomatis menetapkan tanda sistem berikut ke snapshot yang dibuat dengan skrip pra dan pasca.

- Nilai: `aws:dlm:pre-script`; Kunci: `SUCCESS|FAILED`

  Nilai tanda `SUCCESS` menunjukkan bahwa skrip pra berhasil dieksekusi. Nilai tanda `FAILED` menunjukkan bahwa skrip pra gagal dieksekusi.
- Nilai: `aws:dlm:post-script`; Kunci: `SUCCESS|FAILED`

  Nilai tanda `SUCCESS` menunjukkan bahwa skrip pasca berhasil dieksekusi. Nilai tanda `FAILED` menunjukkan bahwa skrip pasca gagal dieksekusi.

Untuk dokumen SSM kustom dan cadangan SAP HANA, Anda dapat menyimpulkan pembuatan snapshot yang konsisten aplikasi yang berhasil jika snapshot ditandai dengan `aws:dlm:pre-script:SUCCESS` dan `aws:dlm:post-script:SUCCESS`.

Selain itu, snapshot konsisten aplikasi yang dibuat menggunakan cadangan VSS secara otomatis ditandai dengan:

- Nilai: `AppConsistent tag`; Kunci: `true|false`

  Nilai tanda `true` menunjukkan bahwa pencadangan VSS berhasil dan bahwa snapshot bersifat konsisten aplikasi. Nilai tanda `false` menunjukkan bahwa pencadangan VSS gagal dan bahwa snapshot tidak bersifat konsisten aplikasi.

Pantau skrip pra dan pasca Amazon Data Lifecycle Manager

CloudWatch Metrik Amazon

Amazon Data Lifecycle Manager menerbitkan CloudWatch metrik berikut saat skrip pra dan pasca gagal dan berhasil dan ketika pencadangan VSS gagal dan berhasil.

- `PreScriptStarted`
- `PreScriptCompleted`
- `PreScriptFailed`

- `PostScriptStarted`
- `PostScriptCompleted`
- `PostScriptFailed`
- `VSSBackupStarted`
- `VSSBackupCompleted`
- `VSSBackupFailed`

Untuk informasi selengkapnya, lihat Memantau kebijakan Data Lifecycle Manager menggunakan CloudWatch.

Amazon EventBridge

Amazon Data Lifecycle Manager memancarkan EventBridge peristiwa Amazon berikut saat skrip pra atau pasca dimulai, berhasil, atau gagal

- `DLM Pre Post Script Notification`

Untuk informasi selengkapnya, lihat Memantau kebijakan Data Lifecycle Manager menggunakan EventBridge.

Buat kebijakan kustom Amazon Data Lifecycle Manager untuk EBS-backed AMI

Prosedur berikut menunjukkan cara menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan siklus hidup EBS-backed AMI.

Topik

- Membuat kebijakan siklus hidup AMI
- Pertimbangan untuk kebijakan siklus hidup AMI
- Sumber daya tambahan

Membuat kebijakan siklus hidup AMI

Gunakan salah satu prosedur berikut ini untuk membuat kebijakan siklus hidup AMI.

Console

Untuk membuat kebijakan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager, lalu pilih Buat kebijakan siklus hidup.
3. Pada layar Pilih jenis kebijakan, pilih kebijakan EBS-backed AMI, lalu pilih Ber ikutnya.
4. Di bagian Sumber daya target, untuk Tanda sumber daya target, pilih tanda sumber daya yang mengidentifikasi volume atau instans yang akan dicadangkan. Kebijakan hanya mencadangkan sumber daya yang memiliki tanda tertentu kunci dan nilai pasangan.
5. Untuk Deskripsi, masukkan deskripsi singkat untuk kebijakan tersebut.
6. Untuk peran IAM, pilih peran IAM yang memiliki izin untuk mengelola AMI dan snapshot dan untuk mendeskripsikan instans. Untuk menggunakan peran default yang disediakan oleh Amazon Data Lifecycle Manager, pilih Peran default. Atau, untuk menggunakan peran IAM kustom yang Anda buat sebelumnya, pilih Pilih peran lain, lalu pilih peran yang akan digunakan.
7. Untuk Tanda kebijakan, tambahkan tanda yang akan diterapkan pada kebijakan siklus hidup. Anda dapat menggunakan tanda ini untuk mengidentifikasi dan mengategorikan kebijakan Anda.
8. Untuk Status kebijakan setelah pembuatan, pilih Aktifkan kebijakan untuk memulai pelaksanaan kebijakan pada waktu yang dijadwalkan berikutnya, atau Nonaktifkan kebijakan untuk mencegah agar kebijakan tidak berjalan. Jika Anda tidak mengaktifkan kebijakan sekarang, kebijakan tidak akan mulai membuat AMI sampai Anda mengaktifkannya secara manual setelah pembuatan.
9. Di bagian Boot ulang instans, menunjukkan apakah instans harus di-boot ulang sebelum pembuatan AMI. Untuk mencegah instans yang ditargetkan di-boot ulang, pilih Tidak. Memilih Tidak dapat menyebabkan masalah konsistensi data. Untuk melakukan boot ulang instans sebelum pembuatan AMI, pilih Ya. Memilih ini memastikan konsistensi data tetapi dapat menyebabkan beberapa instans yang ditargetkan di-boot ulang secara bersamaan.
10. Pilih Berikutnya.
11. Pada layar Konfigurasikan jadwal, konfigurasikan jadwal kebijakan. Kebijakan dapat memiliki sampai maksimal 4 jadwal. Jadwal 1 bersifat wajib. Jadwal 2, 3, dan 4 bersifat opsional. Untuk setiap jadwal kebijakan yang Anda tambahkan, lakukan hal berikut:

a. Dalam bagian Detail jadwal, lakukan hal berikut:

   i. Untuk Nama jadwal, tentukan nama deskriptif untuk jadwal.

   ii. Untuk Frekuensi dan bidang terkait, konfigurasikan interval antara kebijakan yang dijalankan.

   Anda dapat mengonfigurasikan kebijakan yang berjalan sesuai jadwal harian, mingguan, bulanan, atau tahunan. Atau, pilih Ekspresi cron kustom untuk menentukan interval hingga satu tahun. Untuk informasi selengkapnya, lihat ekspresi Cron dan nilai di Panduan EventBridge Pengguna Amazon.

   iii. Untuk Dimulai pada, tentukan waktu di mana pelaksanaan kebijakan dijadwalkan untuk dimulai. Pelaksanaan kebijakan pertama dimulai dengan satu jam setelah waktu yang Anda jadwalkan. Waktu harus dimasukkan dalam format `hh:mm` UTC.

   iv. Untuk Jenis retensi, tentukan kebijakan penyimpanan untuk AMI yang dibuat berdasarkan jadwal.

   Anda dapat mempertahankan snapshot berdasarkan jumlah atau usianya.

   Untuk retensi berbasis jumlah, rentangnya adalah 1 sampai `1000`. Setelah jumlah maksimum tercapai, AMI tertua dibatalkan pendaftarannya saat AMI baru dibuat.

   Untuk retensi berbasis usia, rentangnya adalah 1 hari ke `100` tahun. Setelah masa retensi masing-masing berakhir, AMI akan dibatalkan pendaftarannya.

   > **Note**
   >
   > Semua jadwal harus memiliki jenis retensi yang sama. Anda dapat menentukan jenis retensi hanya untuk Jadwal 1. Jadwal 2, 3, dan 4 mewarisi jenis retensi dari Jadwal 1. Setiap jadwal dapat memiliki jumlah atau periode retensi sendiri.

b. Konfigurasikan penandaan untuk AMI.

   Di bagian Penandaan, lakukan hal berikut ini:

   i. Untuk menyalin semua tanda yang ditentukan pengguna dari instans sumber ke AMI yang dibuat oleh jadwal, pilih Salin tanda dari sumber.

ii. Secara default, AMI yang dibuat oleh jadwal secara otomatis ditandai dengan ID dari instans sumber. Untuk mencegah penandaan otomatis ini terjadi, untuk Tanda variabel, hapus petak `instance-id:$(instance-id)`.

iii. Untuk menentukan tanda tambahan untuk ditetapkan ke AMI yang dibuat oleh jadwal ini, pilih Tambahkan tanda.

c. Konfigurasikan penghentian AMI.

Untuk menghentikan AMI ketika seharusnya tidak digunakan lagi, di bagian penghentian AMI, pilih Aktifkan penghentian AMI untuk jadwal ini, lalu tentukan aturan penghentian AMI. Aturan penghentian AMI menentukan kapan AMI akan dihentikan.

Jika jadwal menggunakan retensi AMI berbasis hitungan, Anda harus menentukan jumlah AMI tertua yang akan dihentikan. Jumlah penghentian harus kurang dari atau sama dengan jumlah retensi AMI jadwal, dan tidak boleh lebih dari 1000. Misalnya, jika jadwal dikonfigurasi untuk mempertahankan maksimum 5 AMI, Anda dapat mengonfigurasi jadwal untuk menghentikan hingga 5 AMI tertua yang lama.

Jika jadwal menggunakan retensi AMI berdasarkan usia, Anda harus menentukan periode setelah AMI tidak digunakan lagi. Jumlah penghentian harus kurang dari atau sama dengan periode retensi AMI jadwal, dan tidak boleh lebih dari 10 tahun (120 bulan, 520 minggu, atau 3650 hari). Misalnya, jika jadwal dikonfigurasi untuk mempertahankan AMI selama 10 hari, Anda dapat mengonfigurasi AMI yang dijadwalkan untuk menghentikan AMI setelah periode hingga 10 hari setelah pembuatan.

d. Konfigurasikan salinan lintas-Wilayah.

Untuk menyalin AMI yang dibuat oleh jadwal ke Wilayah yang berbeda, di bagian sal Cross-Region in, pilih Aktifkan salin lintas wilayah. Anda dapat menyalin AMI hingga tiga Wilayah tambahan di akun Anda. Anda harus menentukan aturan salinan lintas wilayah terpisah untuk setiap Wilayah tujuan.

Untuk setiap tujuan Wilayah, Anda dapat menentukan sebagai berikut:

- Kebijakan penyimpanan untuk salinan AMI. Saat periode retensi berakhir, salinan di Wilayah tujuan secara otomatis dideregistrasi.
- Status enkripsi untuk salinan AMI. Jika snapshot sumber dienkripsi, atau jika enkripsi secara default diaktifkan, snapshot yang disalin dienkripsi. Jika sumber AMI tidak dienkripsi dan enkripsi secara default dinonaktifkan, Anda dapat mengaktifkan

enkripsi secara opsional. Jika Anda tidak menentukan kunci KMS, AMI dienkripsi menggunakan kunci KMS default untuk enkripsi EBS di setiap Wilayah tujuan. Jika Anda menentukan kunci KMS untuk Wilayah tujuan, peran IAM yang dipilih harus memiliki akses ke kunci KMS.

- Aturan penghentian untuk salinan AMI. Ketika periode penghentian berakhir, salinan AMI secara otomatis tidak digunakan lagi. Periode penghentian harus kurang dari atau sama dengan periode penyimpanan salinan, dan tidak boleh lebih dari 10 tahun.
- Apakah akan menyalin semua tanda atau tidak ada tanda dari sumber AMI.

> **Note**
>
> Jangan melebihi jumlah salinan AMI bersamaan per Wilayah.

e. Untuk menambahkan jadwal tambahan, pilih Tambahkan jadwal lain, yang terletak di bagian atas layar. Untuk setiap jadwal tambahan, lengkapi bidang seperti yang dijelaskan sebelumnya dalam topik ini.

f. Setelah Anda menambahkan jadwal yang diperlukan, pilih Tinjau kebijakan.

12. Tinjau ringkasan kebijakan, lalu pilih Buat kebijakan.

> **Note**
>
> Jika Anda mendapatkan kesalahan `Role with name AWSDataLifecycleManagerDefaultRoleForAMIManagement already exists`, lihat Memecahkan masalah Amazon Data Lifecycle Manager untuk informasi selengkapnya.

Command line

Gunakan perintah create-lifecycle-policy untuk membuat kebijakan siklus hidup AMI. Untuk `PolicyType`, tentukan `IMAGE_MANAGEMENT`.

> **Note**
>
> Untuk menyederhanakan sintaksis, contoh berikut menggunakan file JSON, `policyDetails.json`, yang mencakup detail kebijakan.

Contoh 1: Age-based retensi dan penghentian AMI

Contoh ini menciptakan kebijakan siklus hidup AMI yang menciptakan AMI dari semua instans yang memiliki kunci tanda `purpose` dengan nilai `production` tanpa melakukan boot ulang instans yang ditargetkan. Kebijakan ini mencakup satu jadwal yang membuat AMI setiap hari pada pukul `01:00` UTC. Kebijakan mempertahankan AMI selama 2 hari dan menghentikannya setelah 1 hari. Hal ini juga menyalin tanda dari instans sumber ke AMI yang membuatnya.

```
aws dlm create-lifecycle-policy \
    --description "My AMI policy" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRoleForAMIManagement \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "IMAGE_MANAGEMENT",
    "ResourceTypes": [
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key": "purpose",
        "Value": "production"
    }],
    "Schedules": [{
            "Name": "DailyAMIs",
            "TagsToAdd": [{
                "Key": "type",
                "Value": "myDailyAMI"
            }],
            "CreateRule": {
                "Interval": 24,
                "IntervalUnit": "HOURS",
                "Times": [
                    "01:00"
                ]
            },
            RetainRule":{
                "Interval" : 2,
                "IntervalUnit" : "DAYS"
```

```
            },
            DeprecateRule": {
                "Interval" : 1,
                "IntervalUnit" : "DAYS"
            },
            "CopyTags": true
        }
    ],
    "Parameters" : {
        "NoReboot":true
    }
}
```

Jika permintaan berhasil, perintah mengembalikan ID kebijakan yang baru dibuat. Berikut ini adalah output contoh.

```
{
   "PolicyId": "policy-9876543210abcdef0"
}
```

Contoh 2: Count-based retensi dan penghentian AMI dengan salinan lintas wilayah

Contoh ini menciptakan kebijakan siklus hidup AMI yang membuat AMI dari semua instans yang memiliki kunci tanda `purpose` dengan nilai `production` dan melakukan boot ulang instans target. Kebijakan ini mencakup satu jadwal yang membuat AMI setiap 6 jam mulai pukul `17:30` UTC. Kebijakan mempertahankan 3 AMI dan secara otomatis menghentikan 2 AMI tertua. Kebijakan ini juga memiliki aturan menyalin lintas wilayah yang menyalin AMI ke `us-east-1`, mempertahankan 2 salinan AMI, dan secara otomatis menghentikan AMI yang tertua.

```
aws dlm create-lifecycle-policy \
    --description "My AMI policy" \
    --state ENABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRoleForAMIManagement \
    --policy-details file://policyDetails.json
```

Berikut ini adalah contoh file `policyDetails.json`.

```
{
    "PolicyType": "IMAGE_MANAGEMENT",
    "ResourceTypes" : [
```

```
        "INSTANCE"
    ],
    "TargetTags": [{
        "Key":"purpose",
        "Value":"production"
    }],
    "Parameters" : {
          "NoReboot": true
    },
    "Schedules" : [{
        "Name" : "Schedule1",
        "CopyTags": true,
        "CreateRule" : {
            "Interval": 6,
            "IntervalUnit": "HOURS",
            "Times" : ["17:30"]
        },
        "RetainRule":{
            "Count" : 3
        },
        "DeprecateRule":{
            "Count" : 2
        },
        "CrossRegionCopyRules": [{
            "TargetRegion": "us-east-1",
            "Encrypted": true,
            "RetainRule":{
                "IntervalUnit": "DAYS",
                "Interval": 2
            },
            "DeprecateRule":{
                "IntervalUnit": "DAYS",
                "Interval": 1
            },
            "CopyTags": true
        }]
    }]
}
```

Pertimbangan untuk kebijakan siklus hidup AMI

Pertimbangan umum berikut ini berlaku untuk pembuatan kebijakan siklus hidup AMI:

- Kebijakan siklus hidup AMI hanya menargetkan instans atau volume yang berada di Wilayah yang sama dengan kebijakan.
- Operasi pembuatan AMI pertama dimulai dalam waktu satu jam setelah waktu mulai yang ditentukan. Operasi pembuatan AMI selanjutnya dimulai dalam waktu yang dijadwalkan selama satu jam.
- Saat Amazon Data Lifecycle Manager membatalkan pendaftaran AMI, secara otomatis akan menghapusnya untuk mendukung snapshot.
- Tanda sumber daya peka huruf besar dan kecil.
- Jika Anda menghapus tanda target dari instans yang ditargetkan oleh kebijakan, Amazon Data Lifecycle Manager tidak lagi mengelola AMI yang ada dalam standar; Anda harus menghapusnya secara manual jika tidak diperlukan lagi.
- Anda dapat membuat beberapa kebijakan untuk mencadangkan volume atau instans. Misalnya, jika instans memiliki dua tanda, tandai A adalah target kebijakan A untuk membuat AMI setiap 12 jam, dan tanda B adalah target kebijakan B untuk membuat AMI setiap 24 jam, Amazon Data Lifecycle Manager menciptakan AMI sesuai jadwal untuk kedua kebijakan. Atau, Anda dapat mencapai hasil yang sama dengan membuat satu kebijakan yang memiliki beberapa jadwal. Misalnya, Anda dapat membuat kebijakan tunggal yang hanya menargetkan tanda A, dan menentukan dua jadwal — satu untuk setiap 12 jam dan satu untuk setiap 24 jam.
- Volume baru yang dipasang ke instans target setelah kebijakan dibuat secara otomatis disertakan dalam pencadangan saat pelaksanaan kebijakan berikutnya. Semua volume yang dilampirkan pada instans saat pelaksanaan kebijakan disertakan.
- Jika Anda membuat kebijakan dengan jadwal berbasis kron kustom yang dikonfigurasi untuk membuat hanya satu snapshot, kebijakan tidak akan secara otomatis menghapus AMI ketika ambang retensi tercapai. Anda harus menghapus AMI secara manual jika tidak lagi diperlukan.
- Jika Anda membuat kebijakan berbasis usia dengan periode penyimpanan lebih pendek dari frekuensi pembuatan, Amazon Data Lifecycle Manager akan selalu mempertahankan AMI terakhir hingga snapshot berikutnya dibuat. Misalnya, jika kebijakan berbasis usia membuat satu AMI setiap bulan dengan periode retensi tujuh hari, Amazon Data Lifecycle Manager akan mempertahankan setiap AMI selama satu bulan, meskipun periode retensi adalah tujuh hari.
- Untuk kebijakan berbasis jumlah, Amazon Data Lifecycle Manager selalu membuat AMI sesuai dengan frekuensi pembuatan sebelum mencoba membatalkan pendaftaran AMI tertua sesuai dengan kebijakan retensi.
- Diperlukan beberapa jam untuk berhasil membatalkan pendaftaran AMI dan menghapus snapshot dukungan terkait. Jika Amazon Data Lifecycle Manager membuat AMI berikutnya sebelum AMI

yang dibuat sebelumnya berhasil dihapus, Anda dapat mempertahankan sementara jumlah AMI yang lebih besar dari jumlah retensi Anda.

Pertimbangan berikut berlaku untuk mengakhiri instans yang ditargetkan oleh kebijakan:

- Jika Anda menghentikan instans yang ditargetkan oleh kebijakan dengan jadwal penyimpanan berbasis hitungan, kebijakan tersebut tidak lagi mengelola AMI yang sebelumnya dibuat dari instans yang dihentikan. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda menghentikan instans yang ditargetkan oleh kebijakan dengan jadwal penyimpanan berbasis usia, kebijakan tersebut terus membatalkan pendaftaran sampai AMI yang sebelumnya dibuat dari instans yang dihentikan pada jadwal yang ditentukan, hingga, tetapi tidak termasuk, AMI terakhir. Anda harus menghapus AMI secara manual jika tidak lagi diperlukan.

Pertimbangan berikut berlaku untuk kebijakan AMI dan penghentian AMI:

- Jika Anda meningkatkan jumlah penghentian AMI untuk jadwal dengan retensi berbasis hitungan, perubahan akan diterapkan ke semua AMI (yang ada dan baru) yang dibuat oleh jadwal.
- Jika Anda meningkatkan periode penghentian AMI untuk jadwal dengan retensi berbasis usia, perubahan hanya akan diterapkan pada AMI baru. AMI yang ada tidak terpengaruh.
- Jika Anda menghapus aturan penghentian AMI dari jadwal, Amazon Data Lifecycle Manager tidak akan membatalkan penghentian AMI yang sebelumnya tidak digunakan lagi oleh jadwal tersebut.
- Jika mengurangi jumlah atau periode jadwal penghentian AMI, Amazon Data Lifecycle Manager tidak akan membatalkan penghentian AMI yang sebelumnya tidak digunakan lagi oleh jadwal tersebut.
- Jika Anda menghentikan AMI yang dibuat oleh kebijakan AMI secara manual, Amazon Data Lifecycle Manager tidak akan mengganti penghentian tersebut.
- Jika Anda membatalkan penghentian AMI yang sebelumnya dihentikan oleh kebijakan AMI secara manual, Amazon Data Lifecycle Manager tidak akan mengganti penghentian tersebut.
- Jika AMI dibuat oleh beberapa jadwal yang bertentangan, dan satu atau beberapa jadwal tersebut tidak memiliki aturan penghentian AMI, Amazon Data Lifecycle Manager tidak akan menghentikan AMI tersebut.
- Jika AMI dibuat oleh beberapa jadwal yang bertentangan, dan satu atau beberapa jadwal tersebut tidak memiliki aturan penghentian AMI, Amazon Data Lifecycle Manager akan menggunakan aturan penghentian yang menghasilkan tanggal penghentian terbaru.

Pertimbangan berikut berlaku untuk kebijakan AMI dan Recycle Bin:

- Jika Amazon Data Lifecycle Manager membatalkan pendaftaran AMI dan mengirimkannya ke Keranjang Sampah saat ambang retensi kebijakan tercapai, dan Anda memulihkan AMI dari Keranjang Sampah secara manual, Anda harus membatalkan pendaftaran AMI tersebut secara manual saat tidak diperlukan lagi. Amazon Data Lifecycle Manager tidak akan lagi mengelola AMI.
- Jika Anda membatalkan pendaftaran AMI yang dibuat oleh kebijakan secara manual, dan AMI berada di Keranjang Sampah saat ambang retensi kebijakan tercapai, Amazon Data Lifecycle Manager tidak akan membatalkan pendaftaran AMI. Amazon Data Lifecycle Manager tidak mengelola AMI saat berada di Keranjang Sampah.

  Jika snapshot dipulihkan dari Keranjang Sampah sebelum ambang retensi kebijakan tercapai, Amazon Data Lifecycle Manager akan membatalkan pendaftaran AMI tersebut saat ambang retensi kebijakan tercapai.

  Jika AMI dipulihkan dari Keranjang Sampah setelah ambang batas retensi kebijakan tercapai, Amazon Data Lifecycle Manager tidak akan lagi membatalkan pendaftaran AMI tersebut. Anda harus menghapus snapshot AMI secara manual saat tidak lagi diperlukan.

Pertimbangan umum berikut ini berlaku untuk kebijakan AMI dalam status kesalahan:

- Untuk kebijakan dengan jadwal retensi berbasis usia, AMI yang ditetapkan berakhir saat kebijakan berada dalam status `error` akan dipertahankan tanpa batas. Anda harus membatalkan pendaftaran AMI secara manual. Saat Anda mengaktifkan ulang kebijakan, Amazon Data Lifecycle Manager akan melanjutkan pembatalan pendaftaran AMI karena periode retensinya berakhir.
- Untuk kebijakan dengan jadwal retensi berbasis jumlah, kebijakan berhenti membuat dan membatalkan pendaftaran AMI saat berada dalam status `error`. Saat Anda mengaktifkan kembali kebijakan, Amazon Data Lifecycle Manager akan melanjutkan pembuatan AMI, dan melanjutkan pembatalan pendaftaran AMI saat ambang retensi terpenuhi.

Pertimbangan berikut berlaku untuk kebijakan AMI dan menonaktifkan AMI :

- Jika Anda menonaktifkan AMI yang dibuat oleh Amazon Data Lifecycle Manager, dan AMI dinonaktifkan saat ambang retensi tercapai, Amazon Data Lifecycle Manager akan membatalkan pendaftaran AMI dan menghapus snapshot terkait.
- Jika Anda menonaktifkan AMI yang dibuat oleh Amazon Data Lifecycle Manager dan mengarsipkan snapshot terkait secara manual, dan snapshot tersebut diarsipkan saat ambang

retensi terpenuhi, Amazon Data Lifecycle Manager tidak akan menghapus snapshot tersebut dan tidak akan lagi mengelolanya.

Pertimbangan berikut berlaku untuk kebijakan AMI dan perlindungan deregistrasi AMI:

- Jika Anda mengaktifkan perlindungan deregistrasi secara manual untuk AMI yang dibuat oleh Amazon Data Lifecycle Manager, dan masih diaktifkan saat ambang retensi AMI tercapai, Amazon Data Lifecycle Manager tidak lagi mengelola AMI tersebut. Anda harus membatalkan pendaftaran AMI secara manual dan menghapus snapshot yang mendasarinya jika tidak lagi diperlukan.

Sumber daya tambahan

Untuk informasi selengkapnya, lihat blog Men gotomatiskan snapshot Amazon EBS dan manajemen AMI menggunakan blog AWS penyimpanan Amazon Data Lifecycle Manager.

Otomatiskan salinan snapshot lintas akun dengan Data Lifecycle Manager

Mengotomatisasi salinan snapshot lintas akun memungkinkan Anda untuk menyalin snapshot Amazon EBS Anda ke Wilayah tertentu dalam akun terisolasi dan mengenkripsi snapshot tersebut dengan kunci enkripsi. Hal ini memungkinkan Anda melindungi diri dari kehilangan data jika akun Anda disusupi.

Mengotomatisasi salinan snapshot lintas akun melibatkan dua akun:

- Akun sumber—Akun sumber adalah akun yang membuat dan berbagi snapshot dengan akun target. Di akun ini, Anda harus membuat kebijakan snapshot EBS yang membuat snapshot pada interval yang ditetapkan dan kemudian membagikannya dengan akun lain AWS .
- Akun target—Akun target adalah akun dengan akun tujuan tempat snapshot dibagikan, dan itu adalah akun yang membuat salinan dari snapshot bersama. Dalam akun ini, Anda harus membuat salinan lintas akun kebijakan peristiwa yang secara otomatis menyalin snapshot yang dibagi dengan snapshot tersebut oleh satu atau beberapa akun sumber tertentu.

> **Note**
>
> Kebijakan snapshot EBS akun sumber dan kebijakan peristiwa penyalinan lintas akun target harus dibuat di Wilayah yang sama AWS . Akun target kemudian dapat menyalin snapshot ke Wilayah tujuan yang berbeda sesuai kebutuhan.

Topik

- Membuat kebijakan salinan snapshot lintas akun
- Tentukan filter deskripsi snapshot
- Pertimbangan untuk kebijakan penyalinan snapshot lintas akun
- Sumber daya tambahan

Membuat kebijakan salinan snapshot lintas akun

Untuk mempersiapkan akun sumber dan target untuk menyalin snapshot lintas akun, Anda perlu melakukan langkah-langkah berikut:

Langkah 1: Buat kebijakan snapshot EBS (Akun sumber)

Di akun sumber, buat kebijakan snapshot EBS yang akan membuat snapshot dan membagikannya dengan akun target yang diperlukan.

Saat membuat kebijakan, pastikan Anda mengaktifkan berbagi lintas akun dan menentukan AWS akun target yang digunakan untuk berbagi snapshot. Ini adalah akun yang akan digunakan untuk membagikan snapshot. Jika Anda berbagi snapshot terenkripsi, Anda harus memberikan izin akun target yang dipilih untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi volume sumber. Untuk informasi selengkapnya, lihat Langkah 2: Bagikan kunci yang dikelola pelanggan (Akun sumber).

> **Note**
>
> - Buat kebijakan ini di Wil AWS ayah yang sama tempat Anda akan membuat kebijakan peristiwa penyalinan lintas akun target di Langkah 3. Kedua kebijakan harus berada di Wilayah yang sama agar berbagi snapshot lintas akun berfungsi dengan baik.
> - Anda hanya dapat berbagi snapshot yang tidak dienkripsi atau yang dienkripsi menggunakan kunci yang dikelola pelanggan. Anda tidak dapat berbagi snapshot yang

dienkripsi dengan kunci KMS enkripsi EBS default. Jika Anda berbagi snapshot terenkripsi, kemudian Anda juga harus berbagi kunci KMS yang digunakan untuk mengenkripsi volume sumber dengan akun target. Untuk informasi selengkapnya, lihat Mengizinkan pengguna di akun lain untuk menggunakan kunci KMS di Panduan Developer AWS Key Management Service .

Untuk informasi selengkapnya tentang cara membuat kebijakan snapshot EBS, lihat Buat kebijakan kustom Amazon Data Lifecycle Manager untuk snapshot EBS.

Gunakan salah satu metode berikut untuk membuat kebijakan snapshot EBS.

Langkah 2: Bagikan kunci yang dikelola pelanggan (Akun sumber)

Jika Anda berbagi snapshot terenkripsi, Anda harus memberikan izin kepada peran IAM dan akun AWS target (yang Anda pilih di langkah sebelumnya) untuk menggunakan kunci yang dikelola pelanggan yang digunakan untuk mengenkripsi volume sumber.

> **Note**
>
> Lakukan langkah ini hanya jika Anda berbagi snapshot terenkripsi. Jika Anda berbagi snapshot yang tidak dienkripsi, lewati langkah ini.

Console

1. Buka AWS KMS konsol di https://console.aws.amazon.com/kms.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Kunci yang dikelola pelanggan, lalu pilih kunci KMS yang Anda butuhkan untuk berbagi dengan akun target.

   Catat ARN kunci KMS, Anda akan membutuhkan ini nanti.
4. Pada tab Kebijakan kunci, gulir ke bawah ke bagian Pengguna kunci. Pilih Tambahkan, masukkan nama peran IAM yang Anda pilih di langkah sebelumnya, kemudian pilih Tambahkan.
5. Pada tab Kebijakan kunci, gulir ke bawah ke bagian Akun AWS lainnya. Pilih Tambahkan AWS akun lain, lalu tambahkan semua AWS akun target yang Anda pilih untuk berbagi snapshot di langkah sebelumnya.

6. Pilih Simpan perubahan.

Command line

Gunakan perintah get-key-policy untuk mengambil kebijakan kunci yang saat ini dilampirkan pada kunci KMS.

Misalnya, perintah berikut mengambil kebijakan kunci untuk kunci KMS dengan ID `9d5e2b3d-e410-4a27-a958-19e220d83a1e` dan menulisnya ke sebuah file bernama `snapshotKey.json`.

```
$ aws kms get-key-policy \
    --policy-name default \
    --key-id 9d5e2b3d-e410-4a27-a958-19e220d83a1e \
    --query Policy \
    --output text > snapshotKey.json
```

Buka kebijakan kunci menggunakan editor teks pilihan Anda. Menambahkan ARN peran IAM yang Anda tentukan ketika Anda membuat kebijakan snapshot dan ARN akun target yang digunakan untuk berbagi kunci KMS.

Misalnya, dalam kebijakan berikut, ARN dari peran IAM default ditentukan, bersama dengan ARN akun root untuk akun target `222222222222`.

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

```
{
    "Sid" : "Allow use of the key",
    "Effect" : "Allow",
    "Principal" : {
        "AWS" : [
            "arn:aws:iam::111111111111:role/service-role/
AWSDataLifecycleManagerDefaultRole",
```

```
            "arn:aws:iam::222222222222:root"
        ]
    },
    "Action" : [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource" : "*"
},
{
    "Sid" : "Allow attachment of persistent resources",
    "Effect" : "Allow",
    "Principal" : {
        "AWS" : [
            "arn:aws:iam::111111111111:role/service-role/
AWSDataLifecycleManagerDefaultRole",
            "arn:aws:iam::222222222222:root"
        ]
    },
    "Action" : [
        "kms:CreateGrant",
        "kms:ListGrants",
        "kms:RevokeGrant"
    ],
    "Resource" : "*",
    "Condition" : {
        "Bool" : {
          "kms:GrantIsForAWSResource" : "true"
        }
    }
}
```

Simpan dan tutup file . Kemudian gunakan perintah put-key-policy untuk melampirkan kebijakan kunci yang diperbarui untuk bukti kunci KMS.

```
$ aws kms put-key-policy \
    --policy-name default \
    --key-id 9d5e2b3d-e410-4a27-a958-19e220d83a1e \
    --policy file://snapshotKey.json
```

Langkah 3: Buat kebijakan acara salinan lintas akun (Akun target)

Di akun target, Anda harus membuat kebijakan peristiwa penyalinan lintas akun yang secara otomatis akan menyalin snapshot yang dibagikan oleh akun sumber yang diperlukan.

Kebijakan ini hanya berjalan di akun target ketika salah satu akun sumber tertentu berbagi snapshot dengan akun.

> **Note**
>
> Buat kebijakan ini di Wil AWS ayah yang sama dengan kebijakan snapshot EBS akun sumber yang dibuat pada Langkah 1. Kedua kebijakan harus berada di Wilayah yang sama agar berbagi snapshot lintas akun berfungsi dengan baik. Anda kemudian dapat mengonfigurasi kebijakan ini untuk menyalin snapshot ke Wilayah tujuan yang berbeda sesuai kebutuhan.

Gunakan salah satu metode berikut untuk membuat peristiwa penyalinan lintas akun.

Console

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager, lalu pilih Buat kebijakan siklus hidup.
3. Pada layar Pilih jenis kebijakan, pilih sal Cross-account in kebijakan acara, lalu pilih Ber ikutnya.
4. Untuk Deskripsi kebijakan, masukkan deskripsi singkat untuk kebijakan tersebut.
5. Untuk Tanda kebijakan, tambahkan tanda untuk diterapkan ke kebijakan siklus hidup. Anda dapat menggunakan tanda ini untuk mengidentifikasi dan mengategorikan kebijakan Anda.
6. Di bagian Pengaturan peristiwa, tentukan peristiwa berbagi snapshot yang akan menyebabkan kebijakan berjalan. Lakukan hal berikut:

   a. Untuk Berbagi akun, tentukan AWS akun sumber tempat Anda ingin menyalin snapshot bersama. Pilih Tambah akun, masukkan ID AWS akun 12 digit, lalu pilih Tam bah.

   b. Untuk Filter berdasarkan deskripsi, masukkan deskripsi snapshot yang diperlukan menggunakan ekspresi reguler. Hanya snapshot yang dibagikan oleh akun sumber tertentu dan memiliki deskripsi yang cocok dengan filter tertentu yang disalin oleh kebijakan. Untuk informasi selengkapnya, lihat Tentukan filter deskripsi snapshot.

7. Untuk peran IAM, pilih peran IAM yang memiliki izin untuk melakukan tindakan penyalinan snapshot. Untuk menggunakan peran default yang disediakan oleh Amazon Data Lifecycle Manager, pilih Peran default. Atau, untuk menggunakan peran IAM kustom yang Anda buat sebelumnya, pilih Pilih peran lain, lalu pilih peran yang akan digunakan.

   Jika Anda menyalin snapshot terenkripsi, Anda harus memberikan izin peran IAM yang dipilih untuk menggunakan kunci enkripsi KMS yang digunakan untuk mengenkripsi volume sumber. Demikian pula, jika Anda mengenkripsi snapshot di Wilayah tujuan menggunakan kunci KMS yang berbeda, Anda harus memberikan izin kepada peran IAM untuk menggunakan kunci KMS tujuan. Untuk informasi selengkapnya, lihat Langkah 4: Izinkan peran IAM untuk menggunakan kunci KMS yang diperlukan (Akun target).

8. Di bagian Salin tindakan, tentukan tindakan penyalinan snapshot yang harus dilakukan kebijakan saat diaktifkan. Kebijakan ini dapat menyalin snapshot hingga ke tiga Wilayah. Anda harus menentukan aturan salinan lintas wilayah terpisah untuk setiap Wilayah tujuan. Untuk setiap jadwal kebijakan yang Anda tambahkan, lakukan hal berikut:

   a. Untuk Nama, masukkan nama deskriptif untuk tindakan penyalinan.

   b. Untuk Wilayah target, pilih Wilayah untuk menyalin snapshot.

   c. Untuk Kedaluwarsa, tentukan berapa lama untuk mempertahankan salinan snapshot di Wilayah target setelah pembuatan.

   d. Untuk mengenkripsi salinan snapshot, untuk Enkripsi, pilih Aktifkan enkripsi. Jika snapshot sumber dienkripsi, atau jika enkripsi secara default diaktifkan untuk akun Anda, salinan snapshot selalu dienkripsi, meskipun Anda mengaktifkan enkripsi di sini. Jika snapshot sumber tidak dienkripsi dan enkripsi secara default tidak diaktifkan untuk akun Anda, Anda dapat memilih untuk mengaktifkan atau menonaktifkan enkripsi. Jika Anda tidak mengaktifkan enkripsi, tetapi tidak menentukan kunci KMS, snapshot dienkripsi menggunakan kunci KMS enkripsi default untuk setiap Wilayah tujuan. Jika Anda menentukan kunci KMS untuk Wilayah tujuan, Anda harus memiliki akses ke kunci KMS.

9. Untuk menambahkan tindakan penyalinan snapshot tambahan, pilih Tambahkan Wilayah baru.

10. Untuk Status kebijakan setelah pembuatan, pilih Aktifkan kebijakan untuk memulai pelaksanaan kebijakan pada waktu yang dijadwalkan berikutnya, atau Nonaktifkan kebijakan untuk mencegah agar kebijakan tidak berjalan. Jika Anda tidak mengaktifkan kebijakan sekarang, kebijakan tidak akan mulai membuat snapshot sampai Anda mengaktifkannya secara manual setelah pembuatan.

11. Pilih Buat kebijakan.

Command line

Gunakan perintah create-lifecycle-policy untuk membuat kebijakan. Untuk membuat kebijakan peristiwa salinan lintas akun, untuk `PolicyType`, tentukan `EVENT_BASED_POLICY`.

Sebagai contoh, perintah berikut membuat kebijakan menyalin peristiwa lintas akun di akun target `222222222222`. Kebijakan menyalin snapshot yang dibagi oleh akun sumber `111111111111`. Kebijakan menyalin snapshot ke `sa-east-1` dan `eu-west-2`. Snapshot yang disalin ke `sa-east-1` tidak dienkripsi dan dipertahankan selama 3 hari. Snapshot yang disalin ke `eu-west-2` dienkripsi menggunakan kunci KMS `8af79514-350d-4c52-bac8-8985e84171c7` dan dipertahankan selama 1 bulan. Kebijakan menggunakan peran IAM default.

```
$ aws dlm create-lifecycle-policy \
    --description "Copy policy" \
    --state ENABLED \
    --execution-role-arn arn:aws:iam::222222222222:role/service-role/
AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Berikut ini menunjukkan isi file `policyDetails.json`.

```
{
    "PolicyType" : "EVENT_BASED_POLICY",
    "EventSource" : {
        "Type" : "MANAGED_CWE",
        "Parameters": {
            "EventType" : "shareSnapshot",
            "SnapshotOwner": ["111111111111"]
        }
    },
    "Actions" : [{
        "Name" :"Copy Snapshot to Sao Paulo and London",
        "CrossRegionCopy" : [{
            "Target" : "sa-east-1",
             "EncryptionConfiguration" : {
                 "Encrypted" : false
             },
             "RetainRule" : {
             "Interval" : 3,
```

```
            "IntervalUnit" : "DAYS"
            }
        },
        {
            "Target" : "eu-west-2",
            "EncryptionConfiguration" : {
                 "Encrypted" : true,
                 "CmkArn" : "arn:aws:kms:eu-
west-2:222222222222:key/8af79514-350d-4c52-bac8-8985e84171c7"
            },
            "RetainRule" : {
                "Interval" : 1,
                "IntervalUnit" : "MONTHS"
            }
        }]
    }]
}
```

Jika permintaan berhasil, perintah mengembalikan ID kebijakan yang baru dibuat. Berikut ini adalah output contoh.

```
{
    "PolicyId": "policy-9876543210abcdef0"
}
```

Langkah 4: Izinkan peran IAM untuk menggunakan kunci KMS yang diperlukan (Akun target)

Jika Anda menyalin snapshot terenkripsi, Anda harus memberikan izin kepada peran IAM (yang Anda pilih di langkah sebelumnya) untuk menggunakan kunci yang dikelola pelanggan yang digunakan untuk mengenkripsi volume sumber.

> **Note**
>
> Hanya lakukan langkah ini jika Anda menyalin snapshot terenkripsi. Jika Anda menyalin snapshot yang tidak dienkripsi, lewati langkah ini.

Gunakan salah satu metode berikut untuk menambahkan kebijakan yang diperlukan peran IAM.

Console

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran. Cari dan pilih peran IAM yang Anda pilih saat Anda membuat kebijakan peristiwa salinan lintas akun pada langkah sebelumnya. Jika Anda memilih untuk menggunakan peran default, peran tersebut diberi nama AWSDataLifecycleManagerDefaultRole.
3. Pilih Tambahkan kebijakan inline kemudian pilih tab JSON.
4. Ganti kebijakan yang ada dengan yang berikut ini, dan tentukan ARN kunci KMS yang digunakan untuk mengenkripsi volume sumber dan yang dibagikan dengan Anda oleh akun sumber di Langkah 2.

> **Note**
>
> Jika Anda menyalin dari beberapa akun sumber, Anda harus menentukan ARN kunci KMS yang sesuai dari setiap akun sumber.

Pada contoh berikut, kebijakan memberikan izin peran IAM untuk menggunakan kunci KMS `1234abcd-12ab-34cd-56ef-1234567890ab`, yang dibagikan oleh akun sumber `111111111111`, dan kunci KMS`4567dcba-23ab-34cd-56ef-0987654321yz`, yang ada di akun target `222222222222`.

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

JSON

```
{
   "Version":"2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:RevokeGrant",
                "kms:CreateGrant",
                "kms:ListGrants"
            ],
            "Resource": [
                "arn:aws:kms:us-
east-1:111111111111:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                "arn:aws:kms:us-
east-1:222222222222:key/4567dcba-23ab-34cd-56ef-0987654321yz"
            ],
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": [
                "arn:aws:kms:us-
east-1:111111111111:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                "arn:aws:kms:us-
east-1:222222222222:key/4567dcba-23ab-34cd-56ef-0987654321yz"
            ]
        }
    ]
}
```

5. Pilih Tinjau kebijakan
6. Untuk Nama, masukkan nama deskriptif untuk kebijakan, lalu pilih Buat kebijakan.

Command line

Menggunakan editor teks pilihan Anda, buat file JSON baru bernama `policyDetails.json`. Ganti kebijakan yang ada dengan yang berikut ini, dan tentukan ARN kunci KMS yang digunakan untuk mengenkripsi volume sumber dan yang dibagikan dengan Anda oleh akun sumber di Langkah 2.

> **Note**
>
> Jika Anda menyalin dari beberapa akun sumber, Anda harus menentukan ARN kunci KMS yang sesuai dari setiap akun sumber.

Pada contoh berikut, kebijakan memberikan izin peran IAM untuk menggunakan kunci KMS `1234abcd-12ab-34cd-56ef-1234567890ab`, yang dibagikan oleh akun sumber `111111111111`, dan kunci KMS`4567dcba-23ab-34cd-56ef-0987654321yz`, yang ada di akun target `222222222222`.

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "kms:RevokeGrant",
               "kms:CreateGrant",
               "kms:ListGrants"
```

```
            ],
            "Resource": [
                "arn:aws:kms:us-
east-1:111111111111:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                "arn:aws:kms:us-
east-1:222222222222:key/4567dcba-23ab-34cd-56ef-0987654321yz"
            ],
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": [
                "arn:aws:kms:us-
east-1:111111111111:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                "arn:aws:kms:us-
east-1:222222222222:key/4567dcba-23ab-34cd-56ef-0987654321yz"
            ]
        }
    ]
}
```

Simpan dan tutup file. Kemudian, gunakan perintah put-role-policy untuk menambahkan kebijakan terkait peran IAM.

Sebagai contoh

```
$ aws iam put-role-policy \
    --role-name AWSDataLifecycleManagerDefaultRole \
    --policy-name CopyPolicy \
    --policy-document file://AdminPolicy.json
```

Tentukan filter deskripsi snapshot

Ketika Anda membuat kebijakan penyalinan snapshot di akun target, Anda harus menentukan filter deskripsi snapshot. Snapshot deskripsi filter memungkinkan Anda untuk menentukan tingkat tambahan pemfilteran yang memungkinkan Anda mengontrol snapshot yang disalin oleh kebijakan. Artinya, snapshot hanya disalin oleh kebijakan jika dibagikan oleh salah satu akun sumber tertentu, dan memiliki deskripsi snapshot yang cocok dengan filter yang ditentukan. Dengan kata lain, jika snapshot dibagi oleh salah satu akun kursus yang ditentukan, tetapi tidak memiliki deskripsi yang cocok dengan filter yang ditentukan, itu tidak disalin oleh kebijakan.

Deskripsi filter snapshot harus ditentukan menggunakan ekspresi reguler. Ini adalah bidang wajib saat membuat kebijakan peristiwa penyalinan lintas akun menggunakan konsol dan baris perintah. Berikut ini adalah contoh ekspresi reguler yang dapat digunakan:

- `.*`—Filter ini cocok dengan semua deskripsi snapshot. Jika Anda menggunakan ekspresi ini kebijakan akan menyalin semua snapshot yang dibagi oleh salah satu akun sumber tertentu.
- `Created for policy: policy-0123456789abcdef0.*`—Filter ini hanya cocok dengan snapshot yang dibuat oleh kebijakan dengan ID `policy-0123456789abcdef0`. Jika Anda menggunakan ekspresi seperti ini, hanya snapshot yang dibagikan dengan akun Anda oleh salah satu akun sumber tertentu, dan yang telah dibuat oleh kebijakan dengan ID tertentu yang akan disalin oleh kebijakan tersebut.
- `.*production.*`—Filter ini cocok dengan setiap snapshot yang memiliki kata `production` di mana saja dalam deskripsi. Jika Anda menggunakan ekspresi ini kebijakan akan menyalin semua snapshot yang dibagi oleh salah satu akun sumber tertentu dan yang memiliki teks tertentu dalam deskripsi mereka.

Pertimbangan untuk kebijakan penyalinan snapshot lintas akun

Pertimbangan berikut berlaku untuk kebijakan peristiwa penyalinan lintas akun:

- Kebijakan snapshot EBS akun sumber dan kebijakan peristiwa penyalinan lintas akun target harus dibuat di Wilayah yang sama AWS . Setelah snapshot dibagikan, kebijakan akun target dapat menyalin snapshot ke Wilayah tujuan yang berbeda seperti yang ditentukan dalam tindakan penyalinan.
- Anda hanya dapat menyalin snapshot yang tidak dienkripsi atau yang dienkripsi menggunakan kunci yang dikelola pelanggan.

- Anda dapat membuat kebijakan peristiwa penyalinan lintas akun untuk menyalin snapshot yang dibagi di luar Amazon Data Lifecycle Manager.
- Jika Anda ingin mengenkripsi snapshot di akun target, peran IAM yang dipilih untuk kebijakan peristiwa salinan lintas akun harus memiliki izin untuk menggunakan kunci KMS yang diperlukan.

Sumber daya tambahan

Untuk informasi selengkapnya, lihat men gotomatisasi menyalin snapshot Amazon EBS terenkripsi di seluruh blog penyimpanan AWS akun AWS .

Ubah kebijakan Amazon Data Lifecycle Manager

Ingatlah hal-hal berikut saat memodifikasi kebijakan Amazon Data Lifecycle Manager:

- Jika Anda memodifikasi kebijakan AMI atau snapshot dengan menghapus tanda target, volume atau instans tanda tersebut tidak lagi dikelola oleh kebijakan.
- Jika Anda memodifikasi nama jadwal, snapshot atau AMI yang dibuat dengan nama jadwal lama tidak lagi dikelola oleh kebijakan.
- Jika Anda memodifikasi jadwal retensi berbasis usia untuk menggunakan interval waktu baru, interval baru hanya digunakan untuk snapshot baru atau AMI yang dibuat setelah perubahan. Jadwal baru tidak memengaruhi jadwal retensi snapshot atau AMI yang dibuat sebelum perubahan.
- Anda tidak dapat mengubah jadwal penyimpanan kebijakan dari berdasarkan hitungan menjadi berbasis usia setelah pembuatan. Untuk melakukan perubahan ini, Anda harus membuat kebijakan baru.
- Jika Anda menonaktifkan kebijakan dengan jadwal retensi berbasis usia, snapshot, atau AMI yang ditetapkan berakhir saat kebijakan dinonaktifkan akan dipertahankan tanpa batas. Anda harus menghapus snapshot atau membatalkan pendaftaran AMI secara manual. Saat Anda mengaktifkan ulang kebijakan, Amazon Data Lifecycle Manager akan melanjutkan penghapusan snapshot karena periode retensinya berakhir.
- Jika Anda menonaktifkan kebijakan dengan jadwal retensi berbasis jumlah, kebijakan akan berhenti membuat dan menghapus snapshot atau AMI. Saat Anda mengaktifkan kembali kebijakan, Amazon Data Lifecycle Manager akan melanjutkan pembuatan snapshot, dan melanjutkan penghapusan snapshot saat ambang retensi terpenuhi.

- Jika Anda menonaktifkan kebijakan yang memiliki kebijakan yang mengaktifkan pengarsipan snapshot, snapshot yang berada di tingkat arsip pada saat penonaktifan kebijakan tidak lagi dikelola oleh Amazon Data Lifecycle Manager. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda mengaktifkan pengarsipan snapshot pada jadwal berbasis jumlah, aturan pengarsipan berlaku untuk semua snapshot baru yang dibuat dan diarsipkan berdasarkan jadwal, dan juga berlaku untuk snapshot yang ada yang sebelumnya dibuat dan diarsipkan berdasarkan jadwal.
- Jika Anda mengaktifkan pengarsipan snapshot pada jadwal berbasis usia, aturan pengarsipan hanya berlaku untuk snapshot baru yang dibuat setelah mengaktifkan pengarsipan snapshot. Snapshot yang ada yang dibuat sebelum pengaktifan pengarsipan snapshot terus dihapus dari tingkatan penyimpanan masing-masing, sesuai dengan jadwal yang ditetapkan saat snapshot tersebut awalnya dibuat dan diarsipkan.
- Jika Anda menonaktifkan pengarsipan snapshot untuk jadwal berbasis hitungan, jadwal akan segera berhenti mengarsipkan snapshot. Snapshot yang sebelumnya diarsipkan berdasarkan jadwal tetap berada di tingkat arsip dan tidak akan dihapus oleh Amazon Data Lifecycle Manager.
- Jika Anda menonaktifkan pengarsipan snapshot untuk jadwal berdasarkan usia, snapshot yang dibuat oleh kebijakan dan yang dijadwalkan untuk diarsipkan akan dihapus secara permanen pada tanggal dan waktu arsip terjadwal, seperti yang ditunjukkan oleh tanda sistem `aws:dlm:expirationTime`.
- Jika Anda menonaktifkan pengarsipan snapshot untuk jadwal berbasis jumlah, jadwal akan segera berhenti mengarsipkan snapshot. Snapshot yang sebelumnya diarsipkan berdasarkan jadwal tetap berada di tingkat arsip dan tidak akan dihapus oleh Amazon Data Lifecycle Manager.
- Jika Anda mengubah jumlah retensi arsip untuk jadwal berbasis jumlah, jumlah retensi baru menyertakan snapshot yang sudah ada yang sebelumnya diarsipkan oleh jadwal.
- Jika Anda mengubah periode retensi arsip untuk jadwal berdasarkan usia, periode retensi baru hanya berlaku untuk snapshot yang diarsipkan setelah mengubah aturan retensi.

Gunakan salah satu prosedur berikut ini untuk memodifikasi kebijakan siklus hidup.

Console

Untuk mengubah kebijakan siklus hidup

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager.

3. Pilih kebijakan siklus hidup dari daftar.
4. Pilih Tindakan, Modifikasi kebijakan siklus hidup.
5. Ubah pengaturan kebijakan sesuai kebutuhan. Misalnya, Anda dapat mengubah jadwal, menambahkan atau menghapus tanda, atau mengaktifkan atau menonaktifkan kebijakan.
6. Pilih Modifikasi kebijakan.

Command line

Gunakan perintah update-lifecycle-policy untuk mengubah informasi dalam kebijakan siklus hidup. Untuk menyederhanakan sintaksis, contoh ini mengacu pada file JSON, `policyDetailsUpdated.json`, yang mencakup detail kebijakan.

```
aws dlm update-lifecycle-policy \
    --state DISABLED \
    --execution-role-arn
 arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole" \
    --policy-details file://policyDetailsUpdated.json
```

Berikut ini adalah contoh file `policyDetailsUpdated.json`.

```
{
   "ResourceTypes":[
      "VOLUME"
   ],
   "TargetTags":[
      {
         "Key": "costcenter",
         "Value": "120"
      }
   ],
   "Schedules":[
      {
         "Name": "DailySnapshots",
         "TagsToAdd": [
            {
               "Key": "type",
               "Value": "myDailySnapshot"
            }
         ],
         "CreateRule": {
            "Interval": 12,
```

```
            "IntervalUnit": "HOURS",
            "Times": [
               "15:00"
            ]
         },
         "RetainRule": {
            "Count" :5
         },
         "CopyTags": false
      }
   ]
}
```

Untuk melihat kebijakan yang diperbarui, gunakan perintah `get-lifecycle-policy`. Anda dapat melihat bahwa status, nilai tanda, interval snapshot, dan waktu mulai snapshot diubah.

Hapus kebijakan Amazon Data Lifecycle Manager

Ingatlah hal-hal berikut saat menghapus kebijakan Amazon Data Lifecycle Manager:

- Jika Anda menghapus kebijakan, snapshot atau AMI yang dibuat oleh kebijakan tersebut tidak dihapus secara otomatis. Jika Anda tidak lagi membutuhkan snapshot atau AMI, Anda harus menghapusnya secara manual.
- Jika Anda menghapus kebijakan yang mengaktifkan kebijakan pengarsipan snapshot, snapshot yang berada di tingkat arsip pada saat penghapusan kebijakan tidak lagi dikelola oleh Amazon Data Lifecycle Manager. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.
- Jika Anda menghapus kebijakan dengan jadwal berbasis usia yang diaktifkan pengarsipan, snapshot yang dibuat oleh kebijakan dan yang dijadwalkan untuk diarsipkan akan dihapus secara permanen pada tanggal dan waktu arsip terjadwal, seperti yang ditunjukkan oleh tanda sistem `aws:dlm:expirationtime`.

Gunakan salah satu prosedur berikut ini untuk menghapus kebijakan siklus hidup.

Console

Untuk menghapus kebijakan siklus hidup

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

2. Di panel navigasi, pilih Elastic Block Store, Lifecycle Manager.
3. Pilih kebijakan siklus hidup dari daftar.
4. Pilih Tindakan, Hapus kebijakan siklus hidup.
5. Jika diminta untuk mengonfirmasi, pilih Hapus kebijakan.

Command line

Gunakan perintah delete-lifecycle-policy untuk menghapus kebijakan siklus hidup dan mengosongkan tanda target yang ditentukan dalam kebijakan untuk digunakan kembali.

> **Note**
>
> Anda dapat menghapus snapshot yang dibuat hanya oleh Amazon Data Lifecycle Manager.

```
aws dlm delete-lifecycle-policy --policy-id policy-0123456789abcdef0
```

Referensi API Amazon Data Lifecycle Manager memberikan deskripsi dan sintaksis untuk setiap tindakan dan jenis data untuk API Kueri Amazon Data Lifecycle Manager.

Atau, Anda dapat menggunakan salah satu AWS SDK untuk mengakses API dengan cara yang disesuaikan dengan bahasa pemrograman atau platform yang Anda gunakan. Untuk informasi selengkapnya, lihat AWS SDK.

Kontrol akses ke Amazon Data Lifecycle Manager menggunakan IAM

Akses ke Amazon Data Lifecycle Manager memerlukan krekredensial. Kredensial tersebut harus memiliki izin untuk mengakses sumber daya AWS , seperti instans, volume, snapshot, dan AMI.

Izin IAM berikut diperlukan untuk menggunakan Amazon Data Lifecycle Manager.

> **Note**
>
> - Izin `ec2:DescribeAvailabilityZones`, `ec2:DescribeRegions`, `kms:ListAliases`, dan `kms:DescribeKey` diperlukan hanya untuk pengguna konsol. Jika akses konsol tidak diperlukan, Anda dapat menghapus izin.
> - Format ARN dari AWSDataLifecycleManagerDefaultRole peran berbeda tergantung pada apakah itu dibuat menggunakan konsol atau AWS CLI. Jika peran dibuat menggunakan konsol, format ARN adalah `arn:aws:iam::`*`account_id`*`:role/service-role/AWSDataLifecycleManagerDefaultRole`. Jika peran dibuat menggunakan AWS CLI, format ARN adalah`arn:aws:iam::`*`account_id`*`:role/AWSDataLifecycleManagerDefaultRole`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "dlm:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::111122223333:role/
AWSDataLifecycleManagerDefaultRole",
                "arn:aws:iam::111122223333:role/
AWSDataLifecycleManagerDefaultRoleForAMIManagement",
                "arn:aws:iam::111122223333:role/service-role/
AWSDataLifecycleManagerDefaultRole",
                "arn:aws:iam::111122223333:role/service-role/
AWSDataLifecycleManagerDefaultRoleForAMIManagement"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
```

```
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeRegions",
                "kms:ListAliases",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
}
```

Izin untuk enkripsi

Pertimbangkan hal berikut saat bekerja dengan Amazon Data Lifecycle Manager dan sumber daya terenkripsi.

- Jika volume sumber dienkripsi, pastikan peran (AWSDataLifecycleManagerDefaultRoledan AWSDataLifecycleManagerDefaultRoleForAMIManagement) default Amazon Data Lifecycle Manager memiliki izin untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi volume.
- Jika Anda mengaktifkan salin Lintas Wilayah untuk snapshot atau AMI yang tidak dienkripsi yang didukung oleh snapshot tidak terenkripsi, dan memilih untuk mengaktifkan enkripsi di Wilayah tujuan, pastikan bahwa peran default memiliki izin untuk menggunakan kunci KMS yang diperlukan untuk melakukan enkripsi di Wilayah tujuan.
- Jika Anda mengaktifkan salin Lint as Wilayah untuk snapshot terenkripsi atau AMI yang didukung oleh snapshot terenkripsi, pastikan bahwa peran default memiliki izin untuk menggunakan kunci KMS sumber dan tujuan.
- Jika Anda mengaktifkan pengarsipan snapshot untuk snapshot terenkripsi, pastikan peran default Amazon Data Lifecycle Manager (AWSDataLifecycleManagerDefaultRolememiliki izin untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi snapshot.

Untuk informasi selengkapnya, lihat Mengizinkan pengguna di akun lain untuk menggunakan kunci KMS di Panduan Developer AWS Key Management Service .

Untuk informasi selengkapnya, lihat Mengubah izin untuk pengguna pada Panduan Pengguna IAM.

AWS kebijakan terkelola untuk Amazon Data Lifecycle Manager

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS kebijakan terkelola dirancang untuk memberikan izin untuk banyak kasus penggunaan umum. AWS kebijakan terkelola membuatnya lebih efisien bagi Anda untuk menetapkan izin yang sesuai untuk pengguna, grup, dan peran, daripada jika Anda harus menulis kebijakan sendiri.

Namun, Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan ter AWS kelola. AWS sesekali memperbarui izin yang ditentukan dalam kebijakan AWS terkelola. Ketika ini terjadi, pembaruan memengaruhi semua entitas pengguna utama (pengguna, grup, dan peran) yang terkait dengan kebijakan tersebut.

Amazon Data Lifecycle Manager AWS menyediakan kebijakan terkelola untuk kasus penggunaan umum. Kebijakan ini membuatnya lebih efisien untuk menentukan izin yang sesuai dan mengontrol akses ke sumber daya Anda. Kebijakan terkel AWS ola yang disediakan oleh Amazon Data Lifecycle Manager dirancang untuk dilampirkan ke peran yang Anda berikan ke Amazon Data Lifecycle Manager.

Topik

- AWSDataLifecycleManagerServiceRole
- AWSDataLifecycleManagerServiceRoleForAMIManagement
- AWSDataLifecycleManagerSSMFullAccess
- AWS pembaruan kebijakan terkelola

AWSDataLifecycleManagerServiceRole

AWSDataLifecycleManagerServiceRoleKebijakan ini memberikan izin yang sesuai kepada Amazon Data Lifecycle Manager untuk membuat dan mengelola kebijakan snapshot Amazon EBS dan kebijakan peristiwa penyalinan lintas akun.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots",
                "ec2:DeleteSnapshot",
                "ec2:DescribeInstances",
                "ec2:DescribeVolumes",
                "ec2:DescribeSnapshots",
                "ec2:EnableFastSnapshotRestores",
                "ec2:DescribeFastSnapshotRestores",
                "ec2:DisableFastSnapshotRestores",
                "ec2:CopySnapshot",
                "ec2:ModifySnapshotAttribute",
                "ec2:DescribeSnapshotAttribute",
                "ec2:ModifySnapshotTier",
                "ec2:DescribeSnapshotTierStatus",
                "ec2:DescribeAvailabilityZones"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "events:PutRule",
                "events:DeleteRule",
                "events:DescribeRule",
                "events:EnableRule",
                "events:DisableRule",
                "events:ListTargetsByRule",
                "events:PutTargets",
                "events:RemoveTargets"
            ],
            "Resource": "arn:aws:events:*:*:rule/AwsDataLifecycleRule.managed-
cwe.*"
        }
    ]
```

```
}
```

AWSDataLifecycleManagerServiceRoleForAMIManagement

AWSDataLifecycleManagerServiceRoleForAMIManagementKebijakan ini memberikan izin yang sesuai untuk Amazon Data Lifecycle Manager untuk membuat dan mengelola kebijakan Amazon EBS-backed AMI.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": [
                "arn:aws:ec2:*::snapshot/*",
                "arn:aws:ec2:*::image/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages",
                "ec2:DescribeInstances",
                "ec2:DescribeImageAttribute",
                "ec2:DescribeVolumes",
                "ec2:DescribeSnapshots"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DeleteSnapshot",
            "Resource": "arn:aws:ec2:*::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ResetImageAttribute",
```

```
                "ec2:DeregisterImage",
                "ec2:CreateImage",
                "ec2:CopyImage",
                "ec2:ModifyImageAttribute"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:EnableImageDeprecation",
                "ec2:DisableImageDeprecation"
            ],
            "Resource": "arn:aws:ec2:*::image/*"
        }
    ]
}
```

AWSDataLifecycleManagerSSMFullAccess

Memberikan izin kepada Amazon Data Lifecycle Manager untuk melakukan tindakan Systems Manager yang diperlukan untuk menjalankan skrip pra dan pasca di semua instans Amazon EC2.

> **Important**
>
> Kebijakan menggunakan kunci syarat `aws:ResourceTag` untuk membatasi akses ke dokumen SSM tertentu saat menggunakan skrip pra dan pasca. Untuk mengizinkan Amazon Data Lifecycle Manager mengakses dokumen SSM, Anda harus memastikan bahwa dokumen SSM Anda diberi tag. `DLMScriptsAccess:true`

JSON

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

```
                "ssm:GetCommandInvocation",
                "ssm:ListCommands",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowTaggedSSMDocumentsOnly",
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:DescribeDocument",
                "ssm:GetDocument"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/DLMScriptsAccess": "true"
                }
            }
        },
        {
            "Sid": "AllowSpecificAWSOwnedSSMDocuments",
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:DescribeDocument",
                "ssm:GetDocument"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/AWSEC2-CreateVssSnapshot",
                "arn:aws:ssm:*:*:document/AWSSystemsManagerSAP-
CreateDLMSnapshotForSAPHANA"
            ]
        },
        {
            "Sid": "AllowAllEC2Instances",
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
```

```
                "arn:aws:ec2:*:*:instance/*"
            ]
        }
    ]
}
```

AWS pembaruan kebijakan terkelola

AWS layanan memelihara dan memperbarui kebijakan AWS terkelola. Anda tidak dapat mengubah izin dalam kebijakan ter AWS kelola. Layanan terkadang menambahkan izin tambahan ke kebijakan ter AWS kelola untuk mendukung fitur baru. Jenis pembaruan ini akan memengaruhi semua identitas (pengguna, grup, dan peran) di mana kebijakan tersebut dilampirkan. Layanan kemungkinan besar akan memperbarui kebijakan AWS terkelola saat fitur baru diluncurkan atau ketika operasi baru tersedia. Layanan tidak menghapus izin dari kebijakan ter AWS kelola, sehingga pembaruan kebijakan tidak akan merusak izin Anda yang ada.

Tabel berikut memberikan rincian tentang pembaruan kebijakan terkel AWS ola untuk Amazon Data Lifecycle Manager sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan ke umpan RSS pada halaman Riwayat dokumen untuk Panduan Pengguna Amazon EBS.

Ubah	Deskripsi	Date
AWSDataLifecycleManagerServiceRole—Memperbarui izin kebijakan.	Amazon Data Lifecycle Manager menambahkan `ec2:DescribeAvailabilityZones` tindakan untuk memberikan izin kebijakan snapshot untuk mendapatkan informasi tentang Zona Lokal.	Desember 16, 2024

Ubah	Deskripsi	Date
AWSDataLifecycleManagerSSMFullAccess—Memperbarui izin kebijakan.	Memperbarui kebijakan untuk mendukung snapshot yang konsisten dengan aplikasi untuk SAP HANA menggunakan dokumen SSM `AWSSystemsManagerSAP-CreateDLMSnapshotForSAPHANA` .	17 November 2023
AWSDataLifecycleManagerSSMFullAccess—Menambahkan kebijakan terkelola AWS baru.	Amazon Data Lifecycle Manager menambahkan kebijakan AWSDataLifecycleManagerSSMFullAccess AWS terkelola.	7 November 2023

Ubah	Deskripsi	Date
AWSDataLifecycleManagerServiceRole- Menambahkan izin untuk mendukung pengarsipan snapshot.	Amazon Data Lifecycle Manager menambahkan tindakan `ec2:ModifySnapshotTier` dan `ec2:DescribeSnapshotTierStatus` untuk memberikan izin kebijakan snapshot untuk mengarsipkan snapshot dan untuk memeriksa status pengarsipan untuk snapshot.	30 September 2022

Ubah	Deskripsi	Date
AWSDataLifecycleManagerServiceRoleForAMIManagement- Menambahkan izin untuk mendukung penghentian AMI.	Amazon Data Lifecycle Manager menambahkan `ec2:DisableImageDeprecation` tindakan `ec2:EnableImageDeprecation` dan untuk memberikan izin kebijakan EBS-backed AMI untuk mengaktifkan dan menonaktifkan penghentian AMI.	23 Agustus 2021
Amazon Data Lifecycle Manager mulai melacak perubahan	Amazon Data Lifecycle Manager mulai melacak perubahan untuk kebijakan yang AWS dikelolanya.	23 Agustus 2021

Peran layanan IAM untuk Amazon Data Lifecycle Manager

Peran AWS Identity and Access Management (IAM) mirip dengan pengguna, karena itu adalah 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. Peran layanan adalah peran yang diasumsikan AWS oleh layanan untuk melakukan tindakan atas nama Anda. Sebagai layanan yang melakukan operasi pencadangan atas nama Anda, Amazon Data Lifecycle Manager mengharuskan Anda meneruskan peran yang harus diambil saat melakukan operasi kebijakan atas nama Anda. Untuk informasi selengkapnya tentang peran IAM, lihat Peran IAM dalam Panduan Pengguna IAM.

Peran yang diteruskan ke Amazon Data Lifecycle Manager harus memiliki kebijakan IAM dengan izin yang memungkinkan Amazon Data Lifecycle Manager untuk melakukan tindakan yang terkait dengan operasi kebijakan, seperti membuat snapshot dan AMI, menyalin snapshot dan AMI, menghapus snapshot, dan membatalkan pendaftaran AMI. Izin yang berbeda diperlukan untuk setiap jenis kebijakan Amazon Data Lifecycle Manager. Peran ini juga harus memiliki Amazon Data Lifecycle Manager terdaftar sebagai entitas tepercaya, yang memungkinkan Amazon Data Lifecycle Manager untuk mengambil peran.

Topik

- Peran layanan default untuk Amazon Data Lifecycle Manager
- Peran layanan kustom untuk Amazon Data Lifecycle Manager

Peran layanan default untuk Amazon Data Lifecycle Manager

Amazon Data Lifecycle Manager menggunakan peran layanan default berikut:

- AWSDataLifecycleManagerDefaultRole—peran default untuk mengelola snapshot. Peran ini hanya memercayai layanan `dlm.amazonaws.com` untuk mengambil peran dan memungkinkan Amazon Data Lifecycle Manager untuk melakukan tindakan yang diperlukan oleh kebijakan penyalinan snapshot lintas akun dan snapshot atas nama Anda. Peran ini menggunakan kebijakan `AWSDataLifecycleManagerServiceRole` AWS terkelola.

> **Note**
>
> Format ARN peran berbeda tergantung pada apakah itu dibuat menggunakan konsol atau. AWS CLI Jika peran dibuat menggunakan konsol, format ARN adalah `arn:aws:iam::`*`account_id`*`:role/service-role/AWSDataLifecycleManagerDefaultRole`. Jika peran dibuat menggunakan AWS CLI, format ARN adalah`arn:aws:iam::`*`account_id`*`:role/AWSDataLifecycleManagerDefaultRole`.

- AWSDataLifecycleManagerDefaultRoleForAMIManagement—peran default untuk mengelola AMI. Ini hanya mempercayai `dlm.amazonaws.com` layanan untuk mengambil peran dan memungkinkan Amazon Data Lifecycle Manager untuk melakukan tindakan yang diperlukan oleh kebijakan EBS-backed AMI atas nama Anda. Peran ini menggunakan kebijakan `AWSDataLifecycleManagerServiceRoleForAMIManagement` AWS terkelola.

Jika Anda menggunakan konsol Amazon Data Lifecycle Manager, Amazon Data Lifecycle Manager secara otomatis membuat peran AWSDataLifecycleManagerDefaultRole layanan saat pertama kali Anda membuat kebijakan salinan snapshot atau snapshot lintas-akun, dan secara otomatis akan membuat peran AWSDataLifecycleManagerDefaultRoleForAMIManagement layanan saat pertama kali Anda membuat kebijakan EBS-backed AMI.

Jika Anda tidak menggunakan konsol, Anda dapat membuat peran layanan secara manual menggunakan perintah create-default-role. Untuk`--resource-type`, tent `snapshot` ukan untuk membuat AWSDataLifecycleManagerDefaultRole, atau `image` untuk membuat AWSDataLifecycleManagerDefaultRoleForAMIManagement.

```
$ aws dlm create-default-role --resource-type snapshot|image
```

Jika Anda menghapus peran layanan default, kemudian ingin membuatnya lagi, Anda dapat menggunakan proses yang sama untuk membuatnya ulang di akun Anda.

Peran layanan kustom untuk Amazon Data Lifecycle Manager

Sebagai alternatif untuk menggunakan peran layanan default, Anda dapat membuat peran IAM kustom dengan izin yang diperlukan lalu memilihnya saat Anda membuat kebijakan siklus hidup.

Untuk membuat peran IAM kustom

1. Buat peran dengan izin sebagai berikut.

   - Izin diperlukan untuk mengelola kebijakan siklus hidup snapshot

     JSON

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

```
            "Action": [
                "ec2:CreateSnapshot",
                "ec2:CreateSnapshots",
                "ec2:DeleteSnapshot",
                "ec2:DescribeInstances",
                "ec2:DescribeVolumes",
                "ec2:DescribeSnapshots",
                "ec2:EnableFastSnapshotRestores",
                "ec2:DescribeFastSnapshotRestores",
                "ec2:DisableFastSnapshotRestores",
                "ec2:CopySnapshot",
                "ec2:ModifySnapshotAttribute",
                "ec2:DescribeSnapshotAttribute",
                "ec2:ModifySnapshotTier",
                "ec2:DescribeSnapshotTierStatus",
                "ec2:DescribeAvailabilityZones"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "events:PutRule",
                "events:DeleteRule",
                "events:DescribeRule",
                "events:EnableRule",
                "events:DisableRule",
                "events:ListTargetsByRule",
                "events:PutTargets",
                "events:RemoveTargets"
            ],
            "Resource": "arn:aws:events:*:*:rule/
AwsDataLifecycleRule.managed-cwe.*"
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "ssm:GetCommandInvocation",
                "ssm:ListCommands",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:DescribeDocument",
                "ssm:GetDocument"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:document/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/DLMScriptsAccess": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:DescribeDocument",
                "ssm:GetDocument"
            ],
            "Resource": [
                "arn:aws:ssm:*::document/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "StringNotLike": {
                    "aws:ResourceTag/DLMScriptsAccess": "false"
```

```
                }
            }
        }
    ]
}
```

- Izin diperlukan untuk mengelola kebijakan siklus hidup AMI

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": [
                "arn:aws:ec2:*::snapshot/*",
                "arn:aws:ec2:*::image/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages",
                "ec2:DescribeInstances",
                "ec2:DescribeImageAttribute",
                "ec2:DescribeVolumes",
                "ec2:DescribeSnapshots"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DeleteSnapshot",
            "Resource": "arn:aws:ec2:*::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ResetImageAttribute",
                "ec2:DeregisterImage",
                "ec2:CreateImage",
                "ec2:CopyImage",
```

```
                "ec2:ModifyImageAttribute"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:EnableImageDeprecation",
                "ec2:DisableImageDeprecation"
            ],
            "Resource": "arn:aws:ec2:*::image/*"
        }
    ]
}
```

Untuk informasi selengkapnya, lihat Membuat Peran dalam Panduan Pengguna IAM.

2. Tambahkan hubungan kepercayaan ke peran tersebut.

   a. Di konsol IAM, pilih Peran.

   b. Pilih peran yang Anda buat, lalu pilih Hubungan kepercayaan.

   c. Pilih Edit Hubungan Kepercayaan, tambahkan kebijakan berikut, lalu pilih Perbarui Kebijakan Kepercayaan.

      JSON

```
{
 "Version":"2012-10-17",
 "Statement": [{
  "Effect": "Allow",
  "Principal": {
   "Service": "dlm.amazonaws.com"
  },
  "Action": "sts:AssumeRole"
 }]
}
```

      Kami menyarankan Anda menggunakan kunci syarat `aws:SourceAccount` dan `aws:SourceArn` untuk melindungi diri Anda dari masalah wakil yang membingungkan.

Misalnya, Anda dapat menambahkan blok kondisi berikut ke kebijakan kepercayaan sebelumnya. `aws:SourceAccount` adalah pemilik kebijakan siklus hidup dan `aws:SourceArn` adalah ARN dari kebijakan siklus hidup. Jika Anda tidak mengetahui ID kebijakan siklus hidup, Anda dapat mengganti bagian ARN tersebut dengan wildcard (*), lalu memperbarui kebijakan trust setelah Anda membuat kebijakan siklus hidup.

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account_id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:partition:dlm:region:account_id:policy/policy_id"
    }
}
```

Pantau kebijakan Amazon Data Lifecycle Manager

Anda dapat menggunakan fitur berikut untuk siklus hidup snapshot dan AMI.

Fitur

- Konsol dan AWS CLI
- Memantau kebijakan Data Lifecycle Manager menggunakan EventBridge
- Memantau kebijakan Data Lifecycle Manager menggunakan CloudWatch
- Mencatat Panggilan API Amazon Data Lifecycle Manager Menggunakan AWS CloudTrail

Konsol dan AWS CLI

Anda dapat melihat kebijakan siklus hidup Anda menggunakan konsol Amazon EC2 atau AWS CLI. Setiap snapshot dan AMI yang dibuat oleh kebijakan memiliki stempel waktu dan tanda terkait kebijakan. Anda dapat memfilter snapshot dan AMI menggunakan tanda ini untuk memverifikasi bahwa cadangan Anda sedang dibuat seperti yang Anda inginkan.

Memantau kebijakan Data Lifecycle Manager menggunakan EventBridge

Amazon EBS dan Amazon Data Lifecycle Manager memancarkan peristiwa terkait tindakan kebijakan siklus hidup. Anda dapat menggunakan AWS Lambda dan Amazon Ev CloudWatch ents untuk

menangani pemberitahuan acara secara terprogram. Peristiwa dipancarkan atas dasar upaya terbaik. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Peristiwa berikut ini tersedia:

> **Note**
>
> Tidak ada peristiwa yang dipancarkan untuk tindakan kebijakan siklus hidup AMI.

- `createSnapshot` — Peristiwa Amazon EBS yang dipancarkan saat tindakan `CreateSnapshot` berhasil atau gagal. Untuk informasi selengkapnya, lihat Acara EventBridge Amazon untuk Amazon EBS.
- `DLM Policy State Change` — Peristiwa Amazon Data Lifecycle Manager dipancarkan ketika kebijakan siklus hidup memasuki status kesalahan. Peristiwa ini berisi deskripsi penyebab kesalahan.

  Berikut ini adalah contoh peristiwa ketika izin yang diberikan oleh peran IAM tidak memadai.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "DLM Policy State Change",
    source": "aws.dlm",
    "account": "123456789012",
    "time": "2018-05-25T13:12:22Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:dlm:us-east-1:123456789012:policy/policy-0123456789abcdef"
    ],
    "detail": {
        "state": "ERROR",
        "cause": "Role provided does not have sufficient permissions",
        "policy_id": "arn:aws:dlm:us-east-1:123456789012:policy/
policy-0123456789abcdef"
    }
}
```

  Berikut ini adalah contoh peristiwa saat batas terlampaui.

```
{
```

```
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "DLM Policy State Change",
    "source": "aws.dlm",
    "account": "123456789012",
    "time": "2018-05-25T13:12:22Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:dlm:us-east-1:123456789012:policy/policy-0123456789abcdef"
    ],
    "detail":{
        "state": "ERROR",
        "cause": "Maximum allowed active snapshot limit exceeded",
        "policy_id": "arn:aws:dlm:us-east-1:123456789012:policy/
policy-0123456789abcdef"
    }
}
```

- `DLM Pre Post Script Notification` — Peristiwa yang dipancarkan ketika skrip pra atau pasca dimulai, berhasil, atau gagal.

  Berikut ini contoh peristiwa saat cadangan VSS berhasil.

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "DLM Pre Post Script Notification",
    "source": "aws.dlm",
    "account": "123456789012",
    "time": "2023-10-27T22:04:52Z",
    "region": "us-east-1",
    "resources": ["arn:aws:dlm:us-east-1:123456789012:policy/
policy-01234567890abcdef"],
    "detail": {
        "script_stage": "",
        "result": "success",
        "cause": "",
        "policy_id": "arn:aws:dlm:us-east-1:123456789012:policy/
policy-01234567890abcdef",
        "execution_handler": "AWS_VSS_BACKUP",
        "source": "arn:aws:ec2:us-east-1:123456789012:instance/i-01234567890abcdef",
        "resource_type": "EBS_SNAPSHOT",
        "resources": [{
```

```
            "status": "pending",
            "resource_id": "arn:aws:ec2:us-east-1::snapshot/snap-01234567890abcdef",
            "source": "arn:aws:ec2:us-east-1:123456789012:volume/
vol-01234567890abcdef"
        }],
        "request_id": "a1b2c3d4-a1b2-a1b2-a1b2-a1b2c3d4e5f6",
        "start_time": "2023-10-27T22:03:29.370Z",
        "end_time": "2023-10-27T22:04:51.370Z",
        "timeout_time": ""
    }
}
```

Memantau kebijakan Data Lifecycle Manager menggunakan CloudWatch

Anda dapat memantau kebijakan siklus hidup Amazon Data Lifecycle Manager menggunakan CloudWatch, yang mengumpulkan data mentah dan memprosesnya menjadi metrik yang dapat dibaca dan hampir real-time. Anda dapat menggunakan metrik ini untuk melihat dengan tepat berapa banyak snapshot dan AMI Amazon EBS yang dibuat EBS-backed , dihapus, dan disalin oleh kebijakan Anda dari waktu ke waktu. Anda juga dapat mengatur alarm yang memperhatikan ambang batas tertentu dan mengirim notifikasi atau mengambil tindakan saat ambang batas tersebut terpenuhi.

Metrik disimpan selama periode 15 bulan, sehingga Anda dapat mengakses informasi historis dan mendapatkan pemahaman yang lebih baik tentang bagaimana performa kebijakan siklus hidup Anda selama periode yang lama.

Untuk informasi selengkapnya tentang Amazon CloudWatch, lihat Panduan CloudWatch Pengguna Amazon.

Topik

- Metrik yang didukung
- Melihat CloudWatch metrik untuk kebijakan Anda
- Grafik metrik untuk kebijakan Anda
- Membuat CloudWatch alarm untuk kebijakan
- Contoh kasus penggunaan
- Mengelola kebijakan yang melaporkan tindakan gagal

Metrik yang didukung

Metrik Amazon Data Lifecycle Manager berikut disertakan dalam `AWS/EBS` namespace. Metrik berbeda menurut jenis kebijakan.

Semua metrik dapat diukur pada `DLMPolicyId` dimensi. Statistik yang paling berguna adalah `sum` dan `average`, dan satuan ukurannya adalah `count`.

Pilih tab untuk melihat metrik yang didukung oleh jenis kebijakan tersebut.

EBS snapshot policies

Metrik	Deskripsi
`ResourcesTargeted`	Jumlah sumber daya yang ditargetkan oleh tag yang ditentukan dalam kebijakan snapshot atau EBS-backed AMI.
`SnapshotsCreateStarted`	Jumlah tindakan pembuatan snapshot yang diinisiasi oleh kebijakan snapshot. Setiap tindakan direkam hanya sekali, bahkan jika ada banyak percobaan ulang berikutnya. Jika tindakan pembuatan snapshot gagal, Amazon Data Lifecycle Manager mengirimkan metrik `SnapshotsCreateFailed` .
`SnapshotsCreateCompleted`	Jumlah tindakan pembuatan snapshot yang diinisiasi oleh kebijakan snapshot. Ini termasuk percobaan ulang yang berhasil dalam waktu 60 menit dari waktu yang dijadwalkan.
`SnapshotsCreateFailed`	Jumlah snapshot yang tidak dapat dibuat oleh kebijakan snapshot. Ini termasuk percobaan ulang yang tidak berhasil dalam waktu 60 menit dari waktu yang dijadwalkan.
`SnapshotsSharedCompleted`	Jumlah tindakan pembuatan snapshot yang dibagikan di seluruh akun oleh kebijakan snapshot.
`SnapshotsDeleteCompleted`	Jumlah snapshot yang dihapus oleh kebijakan snapshot atau EBS-backed AMI. Metrik ini hanya berlaku untuk snapshot yang dibuat oleh kebijakan. Hal ini tidak berlaku untuk salinan snapshot lintas Wilayah yang dibuat oleh kebijakan.

Metrik	Deskripsi
	Metrik ini mencakup snapshot yang dihapus saat kebijakan EBS-backed AMI membatalkan pendaftaran AMI.
`SnapshotsDeleteFailed`	Jumlah snapshot yang tidak dapat dihapus oleh kebijakan snapshot atau EBS-backed AMI. Metrik ini hanya berlaku untuk snapshot yang dibuat oleh kebijakan. Hal ini tidak berlaku untuk salinan snapshot lintas Wilayah yang dibuat oleh kebijakan. Metrik ini mencakup snapshot yang dihapus saat kebijakan EBS-backed AMI membatalkan pendaftaran AMI.
`SnapshotsCopiedRegionStarted`	Jumlah tindakan penyalinan snapshot lintas wilayah yang diinisiasi oleh kebijakan snapshot.
`SnapshotsCopiedRegionCompleted`	Jumlah tindakan penyalinan snapshot lintas wilayah yang dibuat oleh kebijakan snapshot. Ini termasuk percobaan ulang yang berhasil dalam waktu 24 jam dari waktu yang dijadwalkan.
`SnapshotsCopiedRegionFailed`	Jumlah tindakan penyalinan snapshot lintas wilayah yang tidak dapat dibuat oleh kebijakan snapshot. Ini termasuk percobaan ulang yang berhasil dalam waktu 24 jam dari waktu yang dijadwalkan.
`SnapshotsCopiedRegionDeleteCompleted`	Jumlah salinan snapshot lintas Wilayah yang dihapus, sebagaimana ditetapkan oleh aturan retensi, oleh kebijakan snapshot.
`SnapshotsCopiedRegionDeleteFailed`	Jumlah salinan snapshot lintas Wilayah yang tidak dapat dihapus, sebagaimana ditetapkan oleh aturan retensi, oleh kebijakan snapshot.
`snapshotsArchiveDeletionFailed`	Jumlah snapshot yang diarsipkan yang tidak dapat dihapus dari tingkat arsip oleh kebijakan snapshot.

Metrik	Deskripsi
`snapshots ArchiveSc heduled`	Jumlah snapshot yang dijadwalkan untuk diarsipkan oleh kebijakan snapshot.
`snapshots ArchiveCo mpleted`	Jumlah snapshot yang berhasil diarsipkan oleh kebijakan snapshot.
`snapshots ArchiveFailed`	Jumlah snapshot yang tidak dapat diarsipkan oleh kebijakan snapshot.
`snapshots ArchiveDe letionCom pleted`	Jumlah snapshot yang diarsipkan yang berhasil dihapus dari tingkat arsip oleh kebijakan snapshot.
`PreScript Started`	Jumlah instans saat skrip pra berhasil dimulai. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
`PreScript Completed`	Jumlah instans saat skrip pra berhasil diselesaikan. Metrik dipancark an meskipun skrip pra selesai di luar periode batas waktu yang ditentukan. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
`PreScript Failed`	Jumlah instans saat skrip pra gagal diselesaikan dengan sukses. Metrik dipancarkan meskipun skrip pra selesai di luar periode batas waktu yang ditentukan. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.

Metrik	Deskripsi
PostScriptStarted	Jumlah instans saat skrip pasca berhasil dimulai. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
PostScriptCompleted	Jumlah instans saat skrip pasca berhasil diselesaikan. Metrik dipancarkan bahkan jika skrip pasca selesai di luar periode batas waktu yang ditentukan. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
PostScriptFailed	Jumlah peristiwa saat skrip pasca gagal diselesaikan dengan sukses. Metrik dipancarkan bahkan jika skrip pasca selesai di luar periode batas waktu yang ditentukan. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
VSSBackupStarted	Jumlah instans saat cadangan VSS berhasil dimulai. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
VSSBackupCompleted	Jumlah instans saat cadangan VSS berhasil diselesaikan. Metrik dipancarkan bahkan jika cadangan VSS selesai di luar periode batas waktu. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.
VSSBackupFailed	Jumlah instans saat pencadangan VSS gagal diselesaikan dengan sukses. Metrik dipancarkan bahkan jika cadangan VSS selesai di luar periode batas waktu. Jika percobaan ulang skrip diaktifkan, metrik ini dapat dipancarkan beberapa kali per kebijakan yang dijalankan.

EBS-backed AMI policies

Metrik berikut dapat digunakan dengan kebijakan EBS-backed AMI:

Metrik	Deskripsi
`ResourcesTargeted`	Jumlah sumber daya yang ditargetkan oleh tag yang ditentukan dalam kebijakan snapshot atau EBS-backed AMI.
`SnapshotsDeleteCompleted`	Jumlah snapshot yang dihapus oleh kebijakan snapshot atau EBS-backed AMI. Metrik ini hanya berlaku untuk snapshot yang dibuat oleh kebijakan. Hal ini tidak berlaku untuk salinan snapshot lintas Wilayah yang dibuat oleh kebijakan. Metrik ini mencakup snapshot yang dihapus saat kebijakan EBS-backed AMI membatalkan pendaftaran AMI.
`SnapshotsDeleteFailed`	Jumlah snapshot yang tidak dapat dihapus oleh kebijakan snapshot atau EBS-backed AMI. Metrik ini hanya berlaku untuk snapshot yang dibuat oleh kebijakan. Hal ini tidak berlaku untuk salinan snapshot lintas Wilayah yang dibuat oleh kebijakan. Metrik ini mencakup snapshot yang dihapus saat kebijakan EBS-backed AMI membatalkan pendaftaran AMI.
`SnapshotsCopiedRegionDeleteCompleted`	Jumlah salinan snapshot lintas Wilayah yang dihapus, sebagaimana ditetapkan oleh aturan retensi, oleh kebijakan snapshot.
`SnapshotsCopiedRegionDeleteFailed`	Jumlah salinan snapshot lintas Wilayah yang tidak dapat dihapus, sebagaimana ditetapkan oleh aturan retensi, oleh kebijakan snapshot.

Metrik	Deskripsi
ImagesCreateStarted	Jumlah CreateImage tindakan yang diprakarsai oleh kebijakan EBS-backed AMI.
ImagesCreateCompleted	Jumlah AMI yang dibuat oleh kebijakan EBS-backed AMI.
ImagesCreateFailed	Jumlah AMI yang tidak dapat dibuat oleh kebijakan EBS-backed AMI.
ImagesDeregisterCompleted	Jumlah AMI yang dideregistrasi oleh kebijakan EBS-backed AMI.
ImagesDeregisterFailed	Jumlah AMI yang tidak dapat dideregistrasi oleh kebijakan EBS-backed AMI.
ImagesCopiedRegionStarted	Jumlah tindakan penyalinan lintas wilayah yang diprakarsai oleh kebijakan EBS-backed AMI.
ImagesCopiedRegionCompleted	Jumlah salinan AMI lintas wilayah yang dibuat oleh kebijakan EBS-backed AMI.
ImagesCopiedRegionFailed	Jumlah salinan AMI lintas wilayah yang tidak dapat dibuat oleh kebijakan EBS-backed AMI.

Metrik	Deskripsi
`ImagesCopiedRegionDeregisterCompleted`	Jumlah salinan AMI lintas wilayah yang EBS-backed dibatalkan pendaftaran, sebagaimana ditentukan oleh aturan penyimpanan, oleh kebijakan AMI.
`ImagesCopiedRegionDeregisteredFailed`	Jumlah salinan AMI lintas wilayah yang tidak dapat dibatalkan pendaftaran, sebagaimana ditentukan oleh aturan penyimpanan, oleh kebijakan AMI. EBS-backed
`EnableImageDeprecationCompleted`	Jumlah AMI yang ditandai untuk tidak digunakan lagi oleh kebijakan EBS-backed AMI.
`EnableImageDeprecationFailed`	Jumlah AMI yang tidak dapat ditandai untuk tidak digunakan lagi oleh kebijakan EBS-backed AMI.
`EnableCopiedImageDeprecationCompleted`	Jumlah salinan AMI lintas wilayah yang ditandai untuk tidak digunakan lagi oleh kebijakan AMI. EBS-backed
`EnableCopiedImageDeprecationFailed`	Jumlah salinan AMI lintas wilayah yang tidak dapat ditandai untuk tidak digunakan lagi oleh kebijakan AMI. EBS-backed

Cross-account copy event policies

Metrik berikut ini dapat digunakan dengan kebijakan peristiwa penyalinan lintas akun:

Metrik	Deskripsi
`SnapshotsCopiedAccountStarted`	Jumlah tindakan menyalin snapshot lintas akun yang diinisiasi oleh kebijakan peristiwa penyalinan lintas akun.
`SnapshotsCopiedAccountCompleted`	Jumlah snapshot yang disalin dari akun lain oleh kebijakan peristiwa penyalinan lintas akun. Ini termasuk percobaan ulang yang berhasil dalam waktu 24 jam dari waktu yang dijadwalkan.
`SnapshotsCopiedAccountFailed`	Jumlah snapshot yang tidak dapat disalin dari akun lain oleh kebijakan peristiwa penyalinan lintas akun. Ini termasuk percobaan ulang yang berhasil dalam waktu 24 jam dari waktu yang dijadwalkan.
`SnapshotsCopiedAccountDeleteCompleted`	Jumlah salinan snapshot lintas Wilayah yang dihapus, sebagaimana ditetapkan oleh aturan retensi, berdasarkan kebijakan peristiwa penyalinan lintas akun.
`SnapshotsCopiedAccountDeleteFailed`	Jumlah salinan snapshot lintas Wilayah yang tidak dapat dihapus, sebagaimana ditetapkan oleh aturan retensi, berdasarkan kebijakan peristiwa penyalinan lintas akun.

Melihat CloudWatch metrik untuk kebijakan Anda

Anda dapat menggunakan alat baris perintah Konsol Manajemen AWS atau untuk membuat daftar metrik yang dikirim Amazon Data Lifecycle Manager ke Amazon CloudWatch.

Amazon EC2 console

Untuk melihat metrik menggunakan konsol Amazon EC2

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

2. Di panel navigasi, pilih Lifecycle Manager.
3. Pilih kebijakan di grid lalu pilih tab Pemantauan.

CloudWatch console

Untuk melihat metrik menggunakan CloudWatch konsol Amazon

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Metrik.
3. Pilih namespace EBS, kemudian pilih Metrik Data Lifecycle Manager.

AWS CLI

Untuk mencantumkan semua metrik yang tersedia untuk Amazon Data Lifecycle Manager

Gunakan perintah list-metrics.

```
$ C:\> aws cloudwatch list-metrics \
    --namespace AWS/EBS
```

Membuat daftar semua metrik untuk kebijakan tertentu

Gunakan perintah list-metrics dan tentukan dimensi `DLMPolicyId`.

```
$ C:\> aws cloudwatch list-metrics \
    --namespace AWS/EBS \
    --dimensions Name=DLMPolicyId,Value=policy-abcdef01234567890
```

Untuk mencantumkan satu metrik di semua kebijakan

Gunakan perintah list-metrics dan tentukan opsi `--metric-name`.

```
$ C:\> aws cloudwatch list-metrics \
    --namespace AWS/EBS \
    --metric-name SnapshotsCreateCompleted
```

Grafik metrik untuk kebijakan Anda

Setelah Anda membuat kebijakan, Anda dapat membuka konsol Amazon EC2 dan melihat grafik pemantauan untuk kebijakan tersebut di tab Pemantauan. Setiap grafik didasarkan pada salah satu metrik Amazon EC2 yang tersedia.

Berikut adalah grafik yang tersedia:

- Sumber daya yang ditargetkan (berdasarkan `ResourcesTargeted`)
- Pembuatan snapshot dimulai (berdasarkan `SnapshotsCreateStarted`)
- Pembuatan snapshot selesai (berdasarkan `SnapshotsCreateCompleted`)
- Pembuatan snapshot gagal (berdasarkan `SnapshotsCreateFailed`)
- Pembagian snapshot selesai (berdasarkan `SnapshotsSharedCompleted`)
- Penghapusan snapshot selesai (berdasarkan) `SnapshotsDeleteCompleted`
- Penghapusan snapshot gagal (berdasarkan `SnapshotsDeleteFailed`)
- Penyalinan snapshot lintas Wilayah dimulai (berdasarkan `SnapshotsCopiedRegionStarted`)
- Penyalinan snapshot lintas Wilayah selesai (berdasarkan `SnapshotsCopiedRegionCompleted`)
- Penyalinan snapshot lintas Wilayah gagal (berdasarkan `SnapshotsCopiedRegionFailed`)
- Penghapusan salinan snapshot lintas Wilayah selesai (berdasarkan `SnapshotsCopiedRegionDeleteCompleted`)
- Penghapusan salinan snapshot lintas Wilayah gagal (berdasarkan `SnapshotsCopiedRegionDeleteFailed`)
- Penyalinan snapshot lintas akun dimulai (berdasarkan `SnapshotsCopiedAccountStarted`)
- Salinan snapshot lintas akun selesai (berdasarkan `SnapshotsCopiedAccountCompleted`)
- Penyalinan snapshot lintas akun gagal (berdasarkan `SnapshotsCopiedAccountFailed`)
- Penghapusan salinan lintas akun snapshot selesai (berdasarkan `SnapshotsCopiedAccountDeleteCompleted`)
- Penghapusan salinan snapshot lintas akun gagal (berdasarkan `SnapshotsCopiedAccountDeleteFailed`)
- Pembuatan AMI dimulai (berdasarkan `ImagesCreateStarted`)
- Pembuatan AMI selesai (berdasarkan `ImagesCreateCompleted`)
- Pembuatan AMI gagal (berdasarkan `ImagesCreateFailed`)

- Pembatalan pendaftaran AMI selesai (berdasarkan `ImagesDeregisterCompleted`)
- Pembatalan pendaftaran AMI gagal (berdasarkan `ImagesDeregisterFailed`)
- Penyalinan lintas wilayah AMI dimulai (berdasarkan `ImagesCopiedRegionStarted`)
- Penyalinan lintas wilayah AMI selesai (berdasarkan `ImagesCopiedRegionCompleted`)
- Penyalinan lintas wilayah AMI gagal (berdasarkan `ImagesCopiedRegionFailed`)
- Pembatalan pendaftaran salinan lintas wilayah AMI selesai (berdasarkan `ImagesCopiedRegionDeregisterCompleted`)
- Pembatalan pendaftaran salinan lintas wilayah AMI gagal (berdasarkan `ImagesCopiedRegionDeregisteredFailed`)
- AMI mengaktifkan penghentian selesai (berdasarkan) `EnableImageDeprecationCompleted`
- AMI mengaktifkan penghentian gagal (berdasarkan) `EnableImageDeprecationFailed`
- Salinan lintas wilayah AMI mengaktifkan penghentian selesai (berdasarkan) `EnableCopiedImageDeprecationCompleted`
- AMI Salinan lintas wilayah mengaktifkan penghentian gagal (berdasarkan) `EnableCopiedImageDeprecationFailed`

Membuat CloudWatch alarm untuk kebijakan

Anda dapat membuat CloudWatch alarm yang memantau CloudWatch metrik untuk kebijakan Anda. CloudWatch akan secara otomatis mengirimi Anda pemberitahuan ketika metrik mencapai ambang batas yang Anda tentukan. Anda dapat membuat CloudWatch alarm menggunakan CloudWatch konsol.

Untuk informasi selengkapnya tentang membuat alarm menggunakan CloudWatch konsol, lihat topik berikut di Panduan CloudWatch Pengguna Amazon.

- Membuat CloudWatch Alarm Berdasarkan Ambang Statis
- Membuat CloudWatch Alarm Berdasarkan Deteksi Anomali

Contoh kasus penggunaan

Berikut adalah contoh kasus penggunaan.

Topik

- Contoh 1: ResourcesTargeted metrik
- Contoh 2: SnapshotDeleteFailed metrik
- Contoh 3: SnapshotsCopiedRegionFailed metrik

Contoh 1: ResourcesTargeted metrik

Anda dapat menggunakan `ResourcesTargeted` metrik untuk memantau jumlah total sumber daya yang ditargetkan oleh kebijakan tertentu setiap kali dijalankan. Ini memungkinkan Anda untuk memicu alarm ketika jumlah sumber daya yang ditargetkan di bawah atau di atas ambang batas yang diharapkan.

Misalnya, jika Anda mengharapkan kebijakan harian Anda untuk membuat cadangan tidak lebih dari `50` volume, Anda dapat membuat alarm yang mengirimkan notifikasi email ketika `sum` untuk `ResourcesTargeted` lebih besar dari `50` selama periode 1 jam. Dengan cara ini, Anda dapat memastikan bahwa tidak ada snapshot yang dibuat secara tak terduga dari volume yang salah tag.

Anda dapat menggunakan perintah berikut untuk membuat alarm ini:

```
$ C:\> aws cloudwatch put-metric-alarm \
    --alarm-name resource-targeted-monitor \
    --alarm-description "Alarm when policy targets more than 50 resources" \
    --metric-name ResourcesTargeted \
    --namespace AWS/EBS \
    --statistic Sum \
    --period 3600 \
    --threshold 50 \
    --comparison-operator GreaterThanThreshold \
    --dimensions "Name=DLMPolicyId,Value=policy_id" \
    --evaluation-periods 1 \
    --alarm-actions sns_topic_arn
```

Contoh 2: SnapshotDeleteFailed metrik

Anda dapat menggunakan metrik `SnapshotDeleteFailed` untuk memantau kegagalan menghapus snapshot sesuai aturan retensi snapshot kebijakan.

Misalnya, jika Anda telah membuat kebijakan yang akan menghapus snapshot secara otomatis setiap dua belas jam, Anda dapat membuat alarm yang memberi tahu tim rekayasa Anda ketika `sum` dari `SnapshotDeletionFailed` lebih besar dari `0` selama periode 1 jam. Ini dapat membantu

menyelidiki retensi snapshot yang tidak tepat dan untuk memastikan bahwa biaya penyimpanan Anda tidak meningkat oleh snapshot yang tidak perlu.

Anda dapat menggunakan perintah berikut untuk membuat alarm ini:

```
$ C:\> aws cloudwatch put-metric-alarm \
    --alarm-name snapshot-deletion-failed-monitor \
    --alarm-description "Alarm when snapshot deletions fail" \
    --metric-name SnapshotsDeleteFailed \
    --namespace AWS/EBS \
    --statistic Sum \
    --period 3600 \
    --threshold 0 \
    --comparison-operator GreaterThanThreshold \
    --dimensions "Name=DLMPolicyId,Value=policy_id" \
    --evaluation-periods 1 \
    --alarm-actions sns_topic_arn
```

Contoh 3: SnapshotsCopiedRegionFailed metrik

Gunakan metrik `SnapshotsCopiedRegionFailed` untuk mengidentifikasi kapan kebijakan Anda gagal menyalin snapshot ke Wilayah lain.

Misalnya, jika kebijakan Anda menyalin snapshot di seluruh Wilayah setiap hari, Anda dapat membuat alarm yang mengirimkan SMS ke tim rekayasa Anda ketika `sum` dari `SnapshotCrossRegionCopyFailed` lebih besar dari `0` selama periode 1 jam. Hal ini dapat berguna untuk memverifikasi apakah snapshot berikutnya dalam garis keturunan berhasil disalin oleh kebijakan.

Anda dapat menggunakan perintah berikut untuk membuat alarm ini:

```
$ C:\> aws cloudwatch put-metric-alarm \
    --alarm-name snapshot-copy-region-failed-monitor \
    --alarm-description "Alarm when snapshot copy fails" \
    --metric-name SnapshotsCopiedRegionFailed \
    --namespace AWS/EBS \
    --statistic Sum \
    --period 3600 \
    --threshold 0 \
    --comparison-operator GreaterThanThreshold \
    --dimensions "Name=DLMPolicyId,Value=policy_id" \
```

```
    --evaluation-periods 1 \
    --alarm-actions sns_topic_arn
```

Mengelola kebijakan yang melaporkan tindakan gagal

Untuk informasi selengkapnya tentang apa yang harus dilakukan ketika salah satu kebijakan melaporkan nilai bukan nol yang tidak terduga untuk metrik tindakan gagal, lihat artikel Apa yang harus saya lakukan jika Amazon Data Lifecycle Manager melaporkan tindakan gagal dalam CloudWatch metrik?

Mencatat Panggilan API Amazon Data Lifecycle Manager Menggunakan AWS CloudTrail

Dengan AWS CloudTrail, Anda dapat melacak aktivitas pengguna dan penggunaan API untuk menunjukkan kepatuhan terhadap kebijakan internal dan standar peraturan. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS CloudTrail.

Amazon Data Lifecycle Manager (Amazon Data Lifecycle Manager) terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Amazon Data Lifecycle Manager. CloudTrail menangkap semua panggilan API untuk Amazon Data Lifecycle Manager sebagai peristiwa, termasuk panggilan dari konsol Amazon Data Lifecycle Manager dan dari panggilan kode ke API Amazon Data Lifecycle Manager. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon S3, termasuk acara untuk Amazon Data Lifecycle Manager. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol di Ri wayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke Amazon Data Lifecycle Manager, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan permintaan dibuat, dan detail tambahan.

Untuk mempelajari selengkap CloudTrail nya, lihat Panduan AWS CloudTrail Pengguna.

Informasi Manajer Siklus Hidup Data Amazon di CloudTrail

CloudTrail diaktifkan pada AWS akun Anda saat Anda membuat akun. Saat aktivitas terjadi di Amazon Data Lifecycle Manager, aktivitas tersebut dicatat dalam suatu CloudTrail acara bersama dengan peristiwa AWS layanan lainnya di ri wayat acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat Melihat Acara dengan Ri CloudTrail wayat Acara.

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk Amazon Data Lifecycle Manager, buat jejak. Jejak CloudTrail memungkinkan pengiriman file log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol, jejak ini diterapkan ke semua Wilayah. 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 bertindak berdasarkan data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat:

- Gambaran Umum untuk Membuat Jejak
- CloudTrail Layanan dan Integrasi yang Didukung
- Mengkonfigurasi Pemberitahuan Amazon SNS untuk CloudTrail
- Mener CloudTrail ima File Log dari Beberapa Wilayah dan M CloudTrail enerima File Log dari Beberapa Akun

Semua tindakan Amazon Data Lifecycle Manager dicatat oleh CloudTrail. Misalnya, panggilan ke `DeleteLifecyclePolicy` tindakan `CreateLifecyclePolicy` dan menghasilkan entri dalam file CloudTrail log. Untuk daftar tindakan lengkap, lihat Tind akan.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut ini:

- Pengguna mana yang membuat permintaan.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi lain, lihat Elemen userIdentity CloudTrail .

Memahami Entri File Log Manajer Siklus Hidup Data Amazon

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 apa 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 tidak muncul dalam urutan tertentu.

Example Contoh: CreateLifecyclePolicy

Berikut ini adalah contoh entri CloudTrail log untuk `CreateLifecyclePolicy` tindakan tersebut.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "Root",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:root",
        "accountId": "123456789012",
        "accessKeyId": "AKIAJA2ELRVCPEXAMPLE",
        "userName": "user",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-07-24T18:01:05Z"
            }
        }
    },
    "eventTime": "2018-07-24T18:20:28Z",
    "eventSource": "dlm.amazonaws.com",
    "eventName": "CreateLifecyclePolicy",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "54.240.230.179",
    "userAgent": "console.ec2.amazonaws.com",
    "requestParameters": {
        "ExecutionRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSDataLifecycleManagerServiceRole",
        "PolicyDetails": {
            "ResourceTypes": [
                "VOLUME"
            ],
            "Schedules": [
                {
                    "CreateRule": {
                        "Interval": 12,
                        "IntervalUnit": "HOURS",
                        "Times": [
                            "09:00"
                        ]
                    },
                    "Name": "Default Schedule",
                    "RetainRule": {
```

```
                        "Count": 3
                    },
                    "TagsToAdd": [
                        {
                            "Key": "Name",
                            "Value": "backup-my-volume"
                        }
                    ]
                }
            ],
            "TargetTags": [
                {
                    "Key": "Name",
                    "Value": "my-volume"
                }
            ]
        },
        "Description": "test-cloudtrail",
        "State": "DISABLED"
    },
    "responseElements": {
        "PolicyId": "policy-04ff8755fce0599eb"
    },
    "requestID": "3d714ca6-8f6e-11e8-92a4-35fd765427f0",
    "eventID": "28ab3121-6040-4a40-80c7-ae59b3adf405",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```

Example Contoh: DeleteLifecyclePolicy

Berikut ini adalah contoh entri CloudTrail log untuk `DeleteLifecyclePolicy` tindakan tersebut.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "Root",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:root",
        "accountId": "123456789012",
        "accessKeyId": "AKIAJA2ELRVCPEXAMPLE",
        "userName": "user",
        "sessionContext": {
```

```
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-07-24T18:01:05Z"
            }
        }
    },
    "eventTime": "2018-07-24T19:33:33Z",
    "eventSource": "dlm.amazonaws.com",
    "eventName": "DeleteLifecyclePolicy",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "54.240.230.241",
    "userAgent": "console.ec2.amazonaws.com",
    "requestParameters": {
        "policyId": "policy-04ff8755fce0599eb"
    },
    "responseElements": null,
    "requestID": "73260971-8f78-11e8-a156-598016e53fb2",
    "eventID": "3740f2fb-0d6a-4712-a7ad-eb9f17103fb2",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```

Titik akhir layanan untuk Amazon Data Lifecycle Manager

Endpoint adalah URL yang berfungsi sebagai titik masuk untuk layanan AWS web. Amazon Data Lifecycle Manager mendukung jenis titik akhir berikut:

- Titik akhir IPv4
- Dual-stack Endpoint yang mendukung IPv4 dan IPv6
- Titik akhir FIPS

Saat Anda membuat permintaan, Anda dapat menentukan titik akhir dan Wilayah yang akan digunakan. Jika Anda tidak menentukan titik akhir, titik akhir IPv4 digunakan secara default. Untuk menggunakan tipe titik akhir yang berbeda, Anda harus menentukannya dalam permintaan Anda. Untuk contoh cara melakukannya, lihat Menentukan titik akhir.

Untuk Pengelola Siklus Hidup Data Amazon, lihat titik akhir Amazon Data Lifecycle Manager di Referensi Umum Amazon Web Services.

Topik

- Titik akhir IPv4
- Dual-stack Titik akhir (IPv4 dan IPv6)
- Titik akhir FIPS
- Menentukan titik akhir

Titik akhir IPv4

Titik akhir IPv4 hanya mendukung lalu lintas IPv4. Titik akhir IPv4 tersedia untuk semua Wilayah.

Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Nama titik akhir menggunakan konvensi penamaan berikut:

- dlm. *region*.amazonaws.com

Misalnya, titik akhir IPv4 untuk Wilayah AS Timur (Virginia Utara) adalah`dlm.us-east-1.amazonaws.com`.

Dual-stack Titik akhir (IPv4 dan IPv6)

Dual-stack Endpoint mendukung lalu lintas IPv4 dan IPv6. Dual-stack titik akhir tersedia untuk semua Wilayah.

Untuk menggunakan IPv6, Anda harus menggunakan titik akhir tumpukan ganda. Saat Anda membuat permintaan ke titik akhir tumpukan ganda, URL titik akhir memutuskan ke alamat IPv6 atau IPv4, tergantung pada protokol yang digunakan oleh jaringan dan klien Anda.

Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Dual-stack nama titik akhir menggunakan konvensi penamaan berikut:

- `dlm.`*region*`.api.aws`

Misalnya, titik akhir tumpukan ganda untuk Wilayah AS Timur (Virginia Utara) adalah`dlm.us-east-1.api.aws`.

Titik akhir FIPS

Amazon Data Lifecycle Manager menyediakan titik FIPS-validated akhir dual-stack (IPv4 dan IPv6) untuk Wilayah berikut:

- `us-east-1` — AS Timur (Virginia Utara)
- `us-east-2` — AS Timur (Ohio)
- `us-west-1` — AS Barat (California Utara)
- `us-west-2` — AS Barat (Oregon)
- `ca-central-1` – Kanada (Pusat)
- `ca-west-1`— Kanada Barat (Calgary)

Titik akhir tumpukan ganda FIPS menggunakan konvensi penamaan berikut:. `dlm-fips.`*`region`*`.api.aws` Misalnya, titik akhir tumpukan ganda FIPS untuk Wilayah Timur AS (Virginia Utara) adalah. `dlm-fips.us-east-1.api.aws`

Menentukan titik akhir

Contoh berikut menunjukkan cara menentukan titik akhir untuk Wilayah `US East (N. Virginia)` menggunakan AWS CLI.

- Dual-stack

```
aws dlm create-default-role \
--resource-type snapshot \
--endpoint-url https://dlm.us-east-2.api.aws
```

- IPv4

```
aws dlm create-default-role \
--resource-type snapshot \
--endpoint-url https://dlm.us-east-2.amazonaws.com
```

Membuat koneksi pribadi antara VPC dan Amazon EBS

Anda dapat membuat koneksi pribadi antara VPC dan Amazon EBS dengan membuat antarmuka titik akhir VPC, didukung oleh. AWS PrivateLink Anda dapat mengakses Amazon EBS seolah-

olah berada di VPC Anda, tanpa menggunakan gateway internet, perangkat NAT, koneksi VPN, atau AWS Direct Connect koneksi. Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan Amazon EBS.

Kami membuat antarmuka jaringan endpoint di setiap subnet yang Anda aktifkan untuk titik akhir antarmuka.

Untuk informasi selengkapnya, lihat Ak AWS PrivateLink ses Layanan AWS melalui dalam AWS PrivateLink Panduan.

> **Note**
>
> Amazon Data Lifecycle Manager mendukung titik akhir VPC antarmuka IPv4 untuk semua komersial dan Wil AWS GovCloud (US) ayah, dan titik akhir VPC antarmuka IPv6 hanya untuk Wilayah komersial.

Pertimbangan untuk titik akhir Amazon EBS VPC

Sebelum Anda menyiapkan titik akhir VPC antarmuka untuk Amazon EBS, tinjau Pertim bangan dalam Panduan. AWS PrivateLink

Secara default, akses penuh ke Amazon EBS diizinkan melalui titik akhir. Anda dapat mengontrol akses ke titik akhir antarmuka menggunakan kebijakan titik akhir VPC. Anda dapat melampirkan kebijakan titik akhir ke titik akhir VPC yang mengontrol akses ke Amazon EBS. Kebijakan titik akhir menentukan informasi berikut:

- Kepala sekolah yang dapat melakukan tindakan.
- T ind akan yang bisa dilakukan.
- Sumber daya di mana tindakan dapat dilakukan.

Untuk informasi selengkapnya, lihat Mengendalikan Akses ke Layanan dengan Titik Akhir VPC dalam Panduan Pengguna VPC Amazon.

Berikut ini adalah contoh kebijakan titik akhir untuk Amazon EBS. Saat dilampirkan ke titik akhir, kebijakan ini memberikan izin kepada semua pengguna untuk mendapatkan informasi ringkasan tentang kebijakan Amazon Data Lifecycle Manager.

```
{
```

```
  "Statement": [{
    "Action": "dlm:GetLifecyclePolicies",
    "Effect": "Allow",
    "Principal": "*",
    "Resource": "*"
  }]
}
```

Membuat titik akhir VPC antarmuka untuk Amazon EBS

Anda dapat membuat titik akhir VPC untuk Amazon EBS menggunakan konsol Amazon VPC atau (). AWS Command Line Interface AWS CLI Untuk informasi selengkapnya, lihat Membuat titik akhir VPC di Panduan Pengguna AWS PrivateLink .

Buat titik akhir VPC untuk Amazon EBS menggunakan nama layanan berikut:

- `com.amazonaws.region.dlm`

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API ke Amazon EBS menggunakan nama DNS defaultnya untuk Wilayah, misalnya,`dlm.us-east-1.amazonaws.com`.

Memecahkan masalah Amazon Data Lifecycle Manager

Dokumentasi berikut dapat membantu Anda memecahkan masalah yang mungkin terjadi.

Topik

- Kesalahan: Peran dengan nama sudah ada

Kesalahan: **`Peran dengan nama sudah ada`**

Deskripsi

Anda akan `Role with name AWSDataLifecycleManagerDefaultRole already exists` menemukan `Role with name AWSDataLifecycleManagerDefaultRoleForAMIManagement already exists` kesalahan saat mencoba membuat kebijakan menggunakan konsol.

Penyebab

Format ARN peran berbeda bergantung pada apakah itu dibuat menggunakan konsol atau AWS CLI. Meskipun ARN berbeda, peran menggunakan nama peran yang sama, yang menghasilkan konflik penamaan peran antara konsol dan AWS CLI.

Solusi

Untuk mengatasi masalah ini, lakukan solusi berikut:

1. (Untuk kebijakan snapshot yang diaktifkan hanya untuk skrip pra dan pasca) Lampirkan kebijakan ter AWSDataLifecycleManagerSSMFullAccess AWS kelola secara manual ke peran AWSDataLifecycleManagerDefaultRole IAM. Untuk informasi selengkapnya, lihat Menambahkan izin identitas IAM.
2. Saat membuat kebijakan Amazon Data Lifecycle Manager, untuk peran IAM, pilih Pilih peran lain, lalu pilih salah satu AWSDataLifecycleManagerDefaultRole (untuk kebijakan snapshot), atau AWSDataLifecycleManagerDefaultRoleForAMIManagement (untuk kebijakan AMI).
3. Terus buat kebijakan seperti biasa.

AWS Backup untuk Toko Blok Elastis Amazon

AWS Backup adalah layanan yang dikelola sepenuhnya yang memusatkan dan mengotomatiskan perlindungan data di seluruh lingkungan Anda. AWS Alih-alih mengelola jadwal snapshot dan retensi per sumber daya, Anda menentukan kebijakan cadangan tunggal dan menerapkannya di seluruh akun dan Wilayah. Tandai sumber daya Anda AWS Backup dan lindungi secara otomatis, tanpa perlu konfigurasi per tim. Layanan ini mendukung banyak AWS layanan, termasuk Amazon EBS, Amazon EC2, Amazon S3, Amazon EFS, Amazon RDS, dan Aurora. Akibatnya, satu kebijakan dapat melindungi volume, database, sistem file, dan penyimpanan objek bersama-sama tanpa alat cadangan terpisah per beban kerja.

Untuk Amazon EBS, AWS Backup memberikan snapshot multi-volume yang konsisten dengan crash yang mengoordinasikan pencadangan point-in-time di semua volume yang terpasang ke instans Amazon EC2. Ini penting untuk database dan aplikasi berkelompok yang menjangkau beberapa volume. AWS Backup Pencarian mengambil pemulihan selangkah lebih jauh dengan mengaktifkan pemulihan tingkat item. Anda dapat menemukan dan memulihkan file atau direktori individual dari snapshot Amazon EBS tanpa memulihkan seluruh volume.

AWS Backup juga menambahkan keamanan dan kepatuhan tingkat perusahaan ke snapshot Anda. Vault dengan celah udara secara logis mengisolasi titik pemulihan dari lingkungan produksi Anda. Vault Lock memberlakukan kekekalan. GuardDuty Integrasi Amazon memindai cadangan untuk malware. Backup Audit Manager terus mengevaluasi postur cadangan Anda dan menghasilkan laporan kepatuhan yang siap diaudit.

Untuk memulai, lihat sumber daya berikut:

- AWS Backup Panduan Pengembang
- Buat cadangan multi-volume dan konsisten terhadap crash
- Kembalikan volume Amazon EBS
- Cari dan pulihkan file individual dari snapshot Amazon EBS

Menggunakan API langsung EBS untuk mengakses konten snapshot EBS

Anda dapat menggunakan API langsung Amazon Elastic Block Store (Amazon EBS) untuk membuat snapshot EBS, menulis data secara langsung ke snapshot, membaca data di snapshot, dan mengidentifikasi perbedaan atau perubahan di antara dua snapshot. Jika Anda adalah vendor perangkat lunak independen (ISV) yang menawarkan layanan pencadangan untuk Amazon EBS, API langsung EBS membuat pelacakan perubahan inkremental pada volume EBS Anda melalui snapshot menjadi lebih efisien dan hemat biaya. Hal ini dapat dilakukan tanpa harus membuat volume baru dari snapshot, lalu gunakan instans Amazon Elastic Compute Cloud (Amazon EC2) untuk membandingkan perbedaannya.

Anda dapat membuat snapshot inkremental secara langsung dari data on-premise ke volume EBS dan cloud untuk digunakan dalam pemulihan bencana cepat. Dengan kemampuan untuk menulis dan membaca snapshot, Anda dapat menulis data on-premise ke snapshot EBS selama terjadi bencana. Kemudian setelah pemulihan, Anda dapat mengembalikannya kembali ke AWS atau lokal dari snapshot. Anda tidak perlu lagi membangun dan memelihara mekanisme yang rumit untuk menyalin data ke dan dari Amazon EBS.

Panduan pengguna ini memberikan gambaran umum tentang elemen-elemen yang membentuk API langsung EBS, dan contoh cara menggunakannya secara efektif. Untuk informasi selengkapnya tentang tindakan, jenis data, parameter, dan kesalahan API, lihat referensi API langsung EBS. Untuk informasi selengkapnya tentang Wil AWS ayah, titik akhir, dan kuota layanan yang didukung untuk API langsung EBS, lihat titik akhir dan kuota Amazon EBS di. Referensi Umum AWS

Topik

- Harga API langsung EBS
- Konsep untuk API langsung EBS
- Kontrol akses ke API langsung EBS menggunakan IAM
- Baca snapshot Amazon EBS dengan API langsung EBS
- Tulis snapshot Amazon EBS dengan API langsung EBS
- Hasil enkripsi untuk API langsung EBS
- Gunakan checksum API langsung EBS untuk memvalidasi data snapshot
- Pastikan idempotensi dalam permintaan API StartSnapshot
- Percobaan ulang kesalahan untuk API langsung EBS

- Mengoptimalkan kinerja untuk API langsung EBS
- Titik akhir layanan untuk API langsung EBS
- AWS Contoh kode SDK untuk API langsung EBS
- Membuat koneksi pribadi antara VPC dan API langsung EBS
- Log panggilan API langsung EBS menggunakan AWS CloudTrail
- Pertanyaan yang sering diajukan untuk API langsung EBS

Harga API langsung EBS

Harga API

Harga yang Anda bayar untuk menggunakan API langsung EBS tergantung pada permintaan yang Anda buat. Untuk informasi selengkapnya, lihat harga Amazon EBS.

- ListChangedBlocksdan ListSnapshotBlocks API dikenakan biaya per permintaan. Misalnya, jika Anda membuat 100.000 permintaan ListSnapshotBlocks API di Wilayah yang mengenakan biaya $0,0006 per 1.000 permintaan, Anda akan dikenakan biaya $0,06 ($0,0006 per 1.000 permintaan x 100).
- GetSnapshotBlockdikenakan biaya per blok yang dikembalikan. Misalnya, jika Anda membuat 100.000 permintaan GetSnapshotBlock API di Wilayah yang mengenakan biaya $0,003 per 1.000 blok yang dikembalikan, Anda akan dikenakan biaya $0,30 ($0,003 per 1.000 blok yang dikembalikan x 100).
- PutSnapshotBlockdikenakan biaya per blok yang ditulis. Misalnya, jika Anda membuat 100.000 permintaan PutSnapshotBlock API di Wilayah yang mengenakan biaya $0,006 per 1.000 blok yang ditulis, Anda akan dikenakan biaya $0,60 ($0,006 per 1.000 blok yang ditulis x 100).

Biaya jaringan

Biaya transfer data

Data yang ditransfer langsung antara API langsung EBS dan instans Amazon EC2 di Wilayah yang sama AWS gratis saat menggunakan titik akhir non-FIPS. Untuk informasi selengkapnya, lihat AWS titik akhir layanan. Jika AWS layanan lain berada di jalur transfer data Anda, Anda akan dikenakan biaya pemrosesan data terkait. Layanan ini termasuk, namun tidak terbatas pada, PrivateLink titik akhir, NAT Gateway dan Transit Gateway.

Titik akhir antarmuka VPC

Jika Anda menggunakan API langsung EBS dari instans atau AWS Lambda fungsi Amazon EC2 di subnet pribadi, Anda dapat menggunakan titik akhir antarmuka VPC, alih-alih menggunakan gateway NAT, untuk mengurangi biaya transfer data jaringan. Untuk informasi selengkapnya, lihat Membuat koneksi pribadi antara VPC dan API langsung EBS.

Konsep untuk API langsung EBS

Berikut ini adalah konsep kunci yang harus Anda pahami sebelum memulai dengan API langsung EBS.

Snapshot

Snapshot adalah sarana utama untuk mencadangkan data dari volume EBS Anda. Dengan API langsung EBS, Anda juga dapat mencadangkan data dari disk on-premise ke snapshot. Untuk menghemat biaya penyimpanan, snapshot berikutnya bersifat bertahap, hanya berisi data volume yang berubah sejak snapshot sebelumnya. Untuk informasi selengkapnya, lihat Snapshot Amazon EBS.

> **Note**
>
> API langsung EBS tidak mendukung snapshot publik dan snapshot lokal aktif. AWS Outposts

Blok

Blok adalah fragmen data di dalam snapshot. Setiap snapshot dapat berisi ribuan blok. Semua blok dalam snapshot memiliki ukuran tetap.

Indeks blok

Indeks blok adalah indeks logis dalam satuan blok 512 KiB. Untuk mengidentifikasi indeks blok, bagilah offset logis data dalam volume logis dengan ukuran blok (offset logis data/524288). Offset logis dari data harus disesuaikan dengan 512 KiB.

Token blok

Token blok adalah hash pengidentifikasi dari sebuah blok di dalam sebuah snapshot, dan digunakan untuk menemukan data blok. Token blok yang dikembalikan oleh API langsung EBS bersifat

sementara. Mereka berubah pada stempel waktu kedaluwarsa yang ditentukan untuk mereka, atau jika Anda menjalankan yang lain ListSnapshotBlocks atau ListChangedBlocks meminta snapshot yang sama.

Checksum

Checksum adalah datum berukuran kecil yang berasal dari blok data untuk tujuan mendeteksi kesalahan yang diperkenalkan selama transmisi atau penyimpanan. API langsung EBS menggunakan checksum untuk memvalidasi integritas data. Saat Anda membaca data dari snapshot EBS, layanan menyediakan checksum Base64-encoded SHA256 untuk setiap blok data yang dikirimkan, yang dapat Anda gunakan untuk validasi. Saat Anda menulis data ke snapshot EBS, Anda harus menyediakan checksum SHA256 yang dienkode Base64 untuk setiap blok data yang ditransmisikan. Layanan memvalidasi data yang diterima menggunakan checksum yang disediakan. Untuk informasi selengkapnya, lihat Gunakan checksum API langsung EBS untuk memvalidasi data snapshot dalam panduan ini.

Enkripsi

Enkripsi melindungi data Anda dengan mengubahnya menjadi kode yang tidak terbaca yang hanya dapat diuraikan oleh orang yang memiliki akses ke kunci KMS yang digunakan untuk mengenkripsinya. Anda dapat menggunakan API langsung EBS untuk membaca dan menulis snapshot terenkripsi, tetapi ada beberapa batasan. Untuk informasi selengkapnya, lihat Hasil enkripsi untuk API langsung EBS dalam panduan ini.

Tindakan API

EBS langsung terdiri dari enam tindakan. Ada tiga tindakan baca dan tiga tindakan tulis. Tindakan baca adalah:

- ListSnapshotBlocks— mengembalikan indeks blok dan token blok blok dalam snapshot yang ditentukan
- ListChangedBlocks— mengembalikan indeks blok dan token blok blok yang berbeda antara dua snapshot tertentu dari volume yang sama dan garis keturunan snapshot.
- GetSnapshotBlock— mengembalikan data dalam blok untuk ID snapshot yang ditentukan, indeks blok, dan token blok.

Tindakan tulis adalah:

- StartSnapshot— memulai snapshot, baik sebagai snapshot tambahan dari yang sudah ada atau sebagai snapshot baru. Snapshot yang dimulai tetap dalam status tertunda hingga selesai menggunakan tindakan CompleteSnapshot .
- PutSnapshotBlock— menambahkan data ke snapshot yang dimulai dalam bentuk blok individual. Anda harus menentukan checksum Base64-encoded SHA256 untuk blok data yang dikirimkan. Layanan memvalidasi checksum setelah transmisi selesai. Permintaan gagal jika checksum yang dihitung oleh layanan tidak sesuai dengan yang Anda tentukan.
- CompleteSnapshot— menyelesaikan snapshot yang dimulai yang dalam keadaan tertunda. Snapshot lalu diubah ke status selesai.

Penandatanganan Tanda Tangan Versi 4

Signature Version 4 adalah proses untuk menambahkan informasi otentikasi ke AWS permintaan yang dikirim oleh HTTP. Untuk keamanan, sebagian besar permintaan AWS harus ditandatangani dengan kunci akses, yang terdiri dari ID kunci akses dan kunci akses rahasia. Kedua kunci ini umumnya disebut sebagai kredensial keamanan Anda. Untuk informasi tentang cara mendapatkan kredensial untuk akun Anda, lihat kredensial keamanan AWS.

Jika Anda ingin membuat permintaan HTTP secara manual, Anda harus mempelajari cara menandatanganinya. Saat Anda menggunakan AWS Command Line Interface (AWS CLI) atau salah satu AWS SDK untuk membuat permintaan AWS, alat ini secara otomatis menandatangani permintaan untuk Anda dengan kunci akses yang Anda tentukan saat mengonfigurasi alat. Saat menggunakan alat ini, Anda tidak perlu mempelajari cara menandatangani permintaan diri.

Untuk informasi selengkapnya, lihat AWS Men andatangani permintaan API di Panduan Pengguna IAM.

Kontrol akses ke API langsung EBS menggunakan IAM

Seorang pengguna harus memiliki kebijakan berikut untuk menggunakan API langsung EBS. Untuk informasi selengkapnya, lihat Mengubah izin untuk pengguna.

Untuk informasi selengkapnya tentang sumber daya, tindakan, dan kunci konteks syarat API langsung EBS untuk digunakan dalam kebijakan izin IAM, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon Elastic Block Store di Referensi Otorisasi Layanan.

> **Important**
>
> Berhati-hatilah saat menetapkan kebijakan berikut kepada pengguna. Dengan menetapkan kebijakan ini, Anda dapat memberikan akses kepada pengguna yang ditolak akses ke sumber daya yang sama melalui API Amazon EC2, seperti tindakan CopySnapshot or CreateVolume .

Izin untuk membaca snapshot

Kebijakan berikut memungkinkan API langsung EBS baca digunakan pada semua snapshot di Wilayah tertentu AWS . Dalam kebijakan, ganti *<Region>* dengan Wilayah snapshot.

Kebijakan berikut mengizinkan pembacaan API langsung EBS untuk digunakan pada snapshot dengan tanda nilai kunci tertentu. Dalam kebijakan, ganti *<Key>* dengan nilai kunci tag, dan *<Value>* dengan nilai tag.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ebs:ListSnapshotBlocks",
                "ebs:ListChangedBlocks",
                "ebs:GetSnapshotBlock"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*",
            "Condition": {
                "StringEqualsIgnoreCase": {
                    "aws:ResourceTag/<Key>": "<Value>"
                }
            }
        }
    ]
}
```

Kebijakan berikut mengizinkan semua pembacaan API langsung EBS untuk digunakan pada semua snapshot dalam akun tersebut hanya pada rentang waktu tertentu. Kebijakan ini mengizinkan penggunaan API langsung EBS berdasarkan pada kunci syarat global `aws:CurrentTime`. Dalam kebijakan tersebut, pastikan Anda mengganti rentang tanggal dan waktu yang ditampilkan sesuai rentang tanggal dan waktu untuk kebijakan Anda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ebs:ListSnapshotBlocks",
                "ebs:ListChangedBlocks",
                "ebs:GetSnapshotBlock"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*",
            "Condition": {
                "DateGreaterThan": {
                    "aws:CurrentTime": "2018-05-29T00:00:00Z"
                },
                "DateLessThan": {
                    "aws:CurrentTime": "2020-05-29T23:59:59Z"
                }
            }
        }
    ]
}
```

Untuk informasi selengkapnya, lihat Mengubah izin untuk pengguna pada Panduan Pengguna IAM.

Izin untuk menulis snapshot

Kebijakan berikut memungkinkan penul isan API langsung EBS digunakan pada semua snapshot di Wilayah tertentu AWS . Dalam kebijakan, ganti *<Region>* dengan Wilayah snapshot.

Kebijakan berikut mengizinkan penulisan API langsung EBS untuk digunakan pada snapshot dengan tanda nilai kunci tertentu. Dalam kebijakan, ganti *<Key>* dengan nilai kunci tag, dan *<Value>* dengan nilai tag.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ebs:StartSnapshot",
                "ebs:PutSnapshotBlock",
                "ebs:CompleteSnapshot"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*",
            "Condition": {
                "StringEqualsIgnoreCase": {
                    "aws:ResourceTag/<Key>": "<Value>"
                }
            }
        }
    ]
}
```

Kebijakan berikut mengizinkan seluruh API langsung EBS untuk digunakan. Hal ini juga memungkinkan tindakan `StartSnapshot` hanya jika ID snapshot induk ditentukan. Oleh karena itu, kebijakan ini memblokir kemampuan untuk memulai snapshot baru menggunakan snapshot induk.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ebs:*",
            "Resource": "*",
            "Condition": {
```

```
                "StringEquals": {
                    "ebs:ParentSnapshot": "arn:aws:ec2:*::snapshot/*"
                }
            }
        }
    ]
}
```

Kebijakan berikut mengizinkan seluruh API langsung EBS untuk digunakan. Kebijakan tersebut juga hanya memungkinkan kunci tanda `user` dibuat untuk snapshot baru. Kebijakan ini juga memastikan bahwa pengguna memiliki akses untuk membuat tanda. Tindakan `StartSnapshot` adalah satu-satunya tindakan yang dapat menentukan tanda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ebs:*",
            "Resource": "*",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": "user"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

Kebijakan berikut mengizinkan semua penulisan API langsung EBS untuk digunakan pada semua snapshot dalam akun tersebut hanya pada rentang waktu tertentu. Kebijakan ini mengizinkan penggunaan API langsung EBS berdasarkan pada kunci syarat global `aws:CurrentTime`. Dalam

kebijakan tersebut, pastikan Anda mengganti rentang tanggal dan waktu yang ditampilkan sesuai rentang tanggal dan waktu untuk kebijakan Anda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ebs:StartSnapshot",
                "ebs:PutSnapshotBlock",
                "ebs:CompleteSnapshot"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*",
            "Condition": {
                "DateGreaterThan": {
                    "aws:CurrentTime": "2018-05-29T00:00:00Z"
                },
                "DateLessThan": {
                    "aws:CurrentTime": "2020-05-29T23:59:59Z"
                }
            }
        }
    ]
}
```

Untuk informasi selengkapnya, lihat Mengubah izin untuk pengguna pada Panduan Pengguna IAM.

Izin untuk menggunakan AWS KMS keys

Kebijakan berikut memberikan izin untuk mendekripsi snapshot terenkripsi menggunakan kunci KMS tertentu. Kebijakan ini juga memberikan izin untuk mengenkripsi snapshot baru menggunakan kunci KMS default untuk enkripsi EBS. Dalam kebijakan, ganti *<Region>* dengan Wilayah kunci KMS, *<AccountId>* dengan ID AWS akun kunci KMS, dan *<KeyId>* dengan ID kunci KMS.

> **Note**
>
> Secara default, semua prinsipal di akun memiliki akses ke kunci KMS terkel AWS ola default untuk enkripsi Amazon EBS, dan mereka dapat menggunakannya untuk operasi enkripsi dan dekripsi EBS. Jika Anda menggunakan kunci yang dikelola pelanggan, Anda harus membuat kebijakan kunci baru atau memodifikasi kebijakan kunci yang ada untuk kunci yang dikelola pelanggan untuk memberi pengguna utama akses utama ke kunci yang dikelola pelanggan. Untuk informasi selengkapnya, lihat Kebijakan kunci di AWS KMS di Panduan Developer AWS Key Management Service .

> **Tip**
>
> Untuk mengikuti prinsip hak akses paling rendah, jangan biarkan akses penuh ke `kms:CreateGrant`. Sebagai gantinya, gunakan kunci `kms:GrantIsForAWSResource` kondisi untuk mengizinkan pengguna membuat hibah pada kunci KMS hanya ketika hibah dibuat atas nama pengguna oleh AWS layanan, seperti yang ditunjukkan pada contoh berikut.

Untuk informasi selengkapnya, lihat Mengubah izin untuk pengguna pada Panduan Pengguna IAM.

Baca snapshot Amazon EBS dengan API langsung EBS

Langkah-langkah berikut menjelaskan cara menggunakan API langsung EBS untuk membaca snapshot:

1. Gunakan ListSnapshotBlocks tindakan untuk melihat semua indeks blok dan blok token blok dalam snapshot. Atau gunakan ListChangedBlocks tindakan untuk hanya melihat indeks blok dan token blok blok yang berbeda antara dua snapshot dari volume yang sama dan garis keturunan snapshot. Tindakan ini membantu Anda mengidentifikasi token blok dan indeks blok dari blok yang mungkin ingin Anda dapatkan datanya.
2. Gunakan GetSnapshotBlock tindakan, dan tentukan indeks blok dan token blok dari blok yang ingin Anda dapatkan datanya.

> **Note**
>
> Anda tidak dapat menggunakan API langsung EBS dengan snapshot yang diarsipkan.

Topik berikut menunjukkan cara membaca dan menulis snapshot menggunakan API langsung EBS.

Topik

- Mencantumkan blok dalam snapshot
- Blok daftar yang berbeda antara dua snapshot
- Dapatkan data blok dari snapshot

Mencantumkan blok dalam snapshot

AWS CLI

Contoh perintah list-snapshot-blocks berikut mengembalikan indeks blok dan token blok dari blok yang berada dalam snapshot `snap-0987654321`. Parameter `--starting-block-index` membatasi hasil untuk memblokir indeks yang lebih besar dari `1000`, dan parameter `--max-results` membatasi hasil untuk `100` blok pertama.

```
aws ebs list-snapshot-blocks --snapshot-id snap-0987654321 --starting-block-
index 1000 --max-results 100
```

Contoh respons berikut untuk perintah sebelumnya mencantumkan indeks blok dan blok token dalam snapshot. Gunakan `get-snapshot-block` memerintahkan dan menentukan indeks blok dan token blok dari blok yang ingin Anda dapatkan datanya. Token blok berlaku hingga waktu kedaluwarsa yang tercantum.

```
{
      "Blocks": [
          {
              "BlockIndex": 1001,
              "BlockToken": "AAABAV3/
PNhXOynVdMYHUpPsetaSvjLB1dtIGfbJv5OJ0sX855EzGTWos4a4"
          },
          {
              "BlockIndex": 1002,
```

```
            "BlockToken": "AAABATGQIgwr0WwIuqIMjCA/Sy7e/
YoQFZsHejzGNvjKauzNgzeI13YHBfQB"
        },
        {
            "BlockIndex": 1007,
            "BlockToken": "AAABAZ9CTuQtUvp/
dXqRWw4d07eOgTZ3jvn6hiW30W9duM8MiMw6yQayzF2c"
        },
        {
            "BlockIndex": 1012,
            "BlockToken": "AAABAQdzxhw0rVV6PNmsfo/
YRIxo9JPR85XxPf1BLjg0Hec6pygYr6laE1p0"
        },
        {
            "BlockIndex": 1030,
            "BlockToken": "AAABAaYvPax6mv+iGWLdTUjQtFWouQ7Dqz6nSD9L
+CbXnvpkswA6iDID523d"
        },
        {
            "BlockIndex": 1031,
            "BlockToken": "AAABATgWZC0XcFwUKvTJbUXMiSPg59KVxJGL
+BWBClkw6spzCxJVqDVaTskJ"
        },
        ...
    ],
    "ExpiryTime": 1576287332.806,
    "VolumeSize": 32212254720,
    "BlockSize": 524288
}
```

AWS API

ListSnapshotBlocksContoh permintaan berikut mengembalikan indeks blok dan token blok blok yang ada dalam snapshot`snap-0acEXAMPLEcf41648`. Parameter `startingBlockIndex` membatasi hasil untuk memblokir indeks yang lebih besar dari `1000`, dan parameter `maxResults` membatasi hasil untuk `100` blok pertama.

```
GET /snapshots/snap-0acEXAMPLEcf41648/blocks?maxResults=100&startingBlockIndex=1000
 HTTP/1.1
  Host: ebs.us-east-2.amazonaws.com
  Accept-Encoding: identity
  User-Agent: <User agent parameter>
  X-Amz-Date: 20200617T231953Z
```

```
  Authorization: <Authentication parameter>
```

Contoh respons berikut untuk permintaan sebelumnya mencantumkan indeks blok dan blok token dalam snapshot. Gunakan GetSnapshotBlock tindakan dan tentukan indeks blok dan token blok dari blok yang ingin Anda dapatkan datanya. Token blok berlaku hingga waktu kedaluwarsa yang tercantum.

```
HTTP/1.1 200 OK
  x-amzn-RequestId: d6e5017c-70a8-4539-8830-57f5557f3f27
  Content-Type: application/json
  Content-Length: 2472
  Date: Wed, 17 Jun 2020 23:19:56 GMT
  Connection: keep-alive

  {
      "BlockSize": 524288,
      "Blocks": [
          {
              "BlockIndex": 0,
              "BlockToken": "AAUBAcuWqOCnDNuKle11s7IIX6jp6FYcC/q8oT93913HhvLvA
+3JRrSybp/0"
          },
          {
              "BlockIndex": 1536,
              "BlockToken":
 "AAUBAWudwfmofcrQhGVlLwuRKm2b8ZXPiyrgoykTRC6IU1NbxKWDY1pPjvnV"
          },
          {
              "BlockIndex": 3072,
              "BlockToken":
 "AAUBAV7p6pC5fKAC7TokoNCtAnZhqq27u6YEXZ3MwRevBkDjmMx6iuA6tsBt"
          },
          {
              "BlockIndex": 3073,
              "BlockToken":
 "AAUBAbqt9zpqBUEvtO2HINAfFaWToOwlPjbIsQOlx6JUN/0+iMQl0NtNbnX4"
          },
          ...
      ],
      "ExpiryTime": 1.59298379649E9,
      "VolumeSize": 3
  }
```

Blok daftar yang berbeda antara dua snapshot

Ingatlah hal berikut saat membuat permintaan paginasi untuk mencantumkan daftar blok yang diubah di antara dua snapshot:

- Respons dapat mencakup satu atau beberapa array `ChangedBlocks` yang kosong. Misalnya:
  - Snapshot 1 — snapshot penuh dengan 1000 blok dengan `0` - 999 indeks blok.
  - Snapshot 2 — snapshot inkremental dengan hanya satu blok yang diubah dengan 999 indeks blok.

  Daftar blok yang diubah untuk snapshot ini dengan `StartingBlockIndex = 0` dan `MaxResults = 100` mengembalikan array `ChangedBlocks` yang kosong. Anda harus meminta hasil yang tersisa menggunakan `nextToken` sampai blok yang diubah dikembalikan dalam set hasil kesepuluh, yang mencakup blok dengan indeks blok `900` - 999.
- Respons dapat melewati blok yang tidak tertulis dalam snapshot. Misalnya:
  - Snapshot 1 — snapshot penuh dengan 1000 blok dengan `2000` - 2999 indeks blok.
  - Snapshot 2 — snapshot inkremental dengan hanya satu blok yang diubah dengan `2000` indeks blok.

  Dengan mendaftar blok yang diubah untuk snapshot ini dengan `StartingBlockIndex = 0` dan `MaxResults = 100`, responsnya akan melewati `0` - 1999 indeks blok dan menyertakan `2000` indeks blok. Respons tidak akan menyertakan array `ChangedBlocks` yang kosong.

AWS CLI

Contoh perintah list-changed-blocks berikut mengembalikan indeks blok dan token blok dari blok yang berbeda antara snapshot `snap-1234567890` dan `snap-0987654321`. Parameter `--starting-block-index` membatasi hasil untuk indeks bloks yang lebih besar dari `0`, dan parameter `--max-results` membatasi hasil untuk `500` blok pertama.

```
aws ebs list-changed-blocks --first-snapshot-id snap-1234567890 --second-snapshot-
id snap-0987654321 --starting-block-index 0 --max-results 500
```

Contoh respons berikut untuk perintah sebelumnya menunjukkan bahwa indeks blok 0, 6000, 6001, 6002, dan 6003 berbeda di antara dua snapshot. Selain itu, indeks blok 6001, 6002, dan 6003 hanya ada dalam ID snapshot pertama yang ditentukan, dan tidak dalam ID snapshot kedua karena tidak ada token blok kedua yang tercantum dalam respons.

Gunakan perintah `get-snapshot-block` dan tentukan indeks blok dan token blok dari blok yang ingin Anda dapatkan datanya. Token blok berlaku hingga waktu kedaluwarsa yang tercantum.

```
{
      "ChangedBlocks": [
          {
              "BlockIndex": 0,
              "FirstBlockToken": "AAABAVahm9SO60Dyi0ORySzn2ZjGjW/
KN3uygGlS0QOYWesbzBbDnX2dGpmC",
              "SecondBlockToken":
 "AAABAf8o0o6UFi1rDbSZGIRaCEdDyBu9TlvtCQxxoKV8qrUPQP7vcM6iWGSr"
          },
          {
              "BlockIndex": 6000,
              "FirstBlockToken": "AAABAbYSiZvJ0/
R9tz8suI8dSzecLjN4kkazK8inFXVintPkdaVFLfCMQsKe",
              "SecondBlockToken":
 "AAABAZnqTdzFmKRpsaMAsDxviVqEI/3jJzI2crq2eFDCgHmyNf777elD9oVR"
          },
          {
              "BlockIndex": 6001,
              "FirstBlockToken": "AAABASBpSJ2UAD3PLxJnCt6zun4/
T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR"
          },
          {
              "BlockIndex": 6002,
              "FirstBlockToken": "AAABASqX4/
NWjvNceoyMUljcRd0DnwbSwNnes1UkoP62CrQXvn47BY5435aw"
          },
          {
              "BlockIndex": 6003,
              "FirstBlockToken":
 "AAABASmJ0O5JxAOce25rF4P1sdRtyIDsX12tFEDunnePYUKOf4PBROuICb2A"
          },
          ...
      ],
      "ExpiryTime": 1576308931.973,
      "VolumeSize": 32212254720,
      "BlockSize": 524288,
      "NextToken": "AAADARqElNng/sV98CYk/bJDCXeLJmLJHnNSkHvLzVaO0zsPH/QM3Bi3zF//
O6Mdi/BbJarBnp8h"
  }
```

AWS API

ListChangedBlocksContoh permintaan berikut mengembalikan indeks blok dan token blok blok yang berbeda antara snapshot `snap-0acEXAMPLEcf41648` dan. `snap-0c9EXAMPLE1b30e2f` Parameter `startingBlockIndex` membatasi hasil untuk memblokir indeks yang lebih besar dari `0`, dan parameter `maxResults` membatasi hasil untuk `500` blok pertama.

```
GET /snapshots/snap-0c9EXAMPLE1b30e2f/changedblocks?
firstSnapshotId=snap-0acEXAMPLEcf41648&maxResults=500&startingBlockIndex=0 HTTP/1.1
  Host: ebs.us-east-2.amazonaws.com
  Accept-Encoding: identity
  User-Agent: <User agent parameter>
  X-Amz-Date: 20200617T232546Z
  Authorization: <Authentication parameter>
```

Contoh respons berikut untuk perintah sebelumnya menunjukkan bahwa indeks blok `0`, `3072`, `6002`, dan `6003` berbeda di antara dua snapshot. Selain itu, indeks blok `6002` dan `6003` hanya ada dalam ID snapshot pertama yang ditentukan, dan tidak dalam ID snapshot kedua karena tidak ada token blok kedua yang tercantum dalam respons.

Gunakan tindakan `GetSnapshotBlock`, dan tentukan indeks blok serta token blok dari blok yang ingin Anda dapatkan datanya. Token blok berlaku hingga waktu kedaluwarsa yang tercantum.

```
HTTP/1.1 200 OK
  x-amzn-RequestId: fb0f6743-6d81-4be8-afbe-db11a5bb8a1f
  Content-Type: application/json
  Content-Length: 1456
  Date: Wed, 17 Jun 2020 23:25:47 GMT
  Connection: keep-alive

  {
      "BlockSize": 524288,
      "ChangedBlocks": [
          {
              "BlockIndex": 0,
              "FirstBlockToken": "AAUBAVaWqOCnDNuKle11s7IIX6jp6FYcC/
tJuVT1GgP23AuLntwiMdJ+OJkL",
              "SecondBlockToken": "AAUBASxzy0Y0b33JVRLoYm3NOresCxn5RO+HVFzXW3Y/
RwfFaPX2Edx8QHCh"
          },
          {
              "BlockIndex": 3072,
```

```
            "FirstBlockToken":
 "AAUBAcHp6pC5fKAC7TokoNCtAnZhqq27u6fxRfZOLEmeXLmHBf2R/Yb24MaS",
            "SecondBlockToken":
 "AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDi3ljDFiytUxBLXYgTmkid"
         },
         {
            "BlockIndex": 6002,
            "FirstBlockToken": "AAABASqX4/
NWjvNceoyMUljcRd0DnwbSwNnes1UkoP62CrQXvn47BY5435aw"
         },
         {
            "BlockIndex": 6003,
            "FirstBlockToken":
 "AAABASmJ0O5JxAOce25rF4P1sdRtyIDsX12tFEDunnePYUKOf4PBROuICb2A"
         },
         ...
     ],
     "ExpiryTime": 1.592976647009E9,
     "VolumeSize": 3
 }
```

Dapatkan data blok dari snapshot

AWS CLI

Contoh perintah get-snapshot-block berikut mengembalikan data dalam indeks blok `6001` dengan token blok `AAABASBpSJ2UAD3PLxJnCt6zun4/T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR`, dalam snapshot `snap-1234567890`. Data biner adalah output file `data` dalam direktori `C:\Temp` pada komputer Windows. Jika Anda menjalankan perintah di komputer Linux atau Unix, ganti jalur output dengan `/tmp/data` untuk mengeluarkan data ke file `data` dalam direktori `/tmp`.

```
aws ebs get-snapshot-block --snapshot-id snap-1234567890 --block-index 6001 --block-
token AAABASBpSJ2UAD3PLxJnCt6zun4/T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR C:/Temp/data
```

Contoh respons berikut untuk perintah sebelumnya menunjukkan ukuran data yang dikembalikan, checksum untuk memvalidasi data, dan algoritma checksum. Data biner secara otomatis disimpan ke direktori dan file yang Anda tentukan dalam perintah permintaan.

```
{
      "DataLength": "524288",
```

```
    "Checksum": "cf0Y6/Fn0oFa4VyjQPOa/iD0zhTflPTKzxGv2OKowXc=",
    "ChecksumAlgorithm": "SHA256"
}
```

AWS API

Contoh GetSnapshotBlock berikut meminta mengembalikan data dalam indeks blok `3072` dengan token blok `AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDi3ljDFiytUxBLXYgTmkid`, dalam snapshot `snap-0c9EXAMPLE1b30e2f`.

```
GET /snapshots/snap-0c9EXAMPLE1b30e2f/blocks/3072?
blockToken=AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDi3ljDFiytUxBLXYgTmkid HTTP/1.1
  Host: ebs.us-east-2.amazonaws.com
  Accept-Encoding: identity
  User-Agent: <User agent parameter>
  X-Amz-Date: 20200617T232838Z
  Authorization: <Authentication parameter>
```

Contoh respons berikut untuk perintah sebelumnya menunjukkan ukuran data yang dikembalikan, checksum untuk memvalidasi data, dan algoritma checksum. Data biner ditransmisikan dalam tubuh respons dan direpresentasikan *`BlockData`* seperti pada contoh berikut.

```
HTTP/1.1 200 OK
  x-amzn-RequestId: 2d0db2fb-bd88-474d-a137-81c4e57d7b9f
  x-amz-Data-Length: 524288
  x-amz-Checksum: Vc0yY2j3qg8bUL9I6GQuI2orTudrQRBDMIhcy7bdEsw=
  x-amz-Checksum-Algorithm: SHA256
  Content-Type: application/octet-stream
  Content-Length: 524288
  Date: Wed, 17 Jun 2020 23:28:38 GMT
  Connection: keep-alive

  BlockData
```

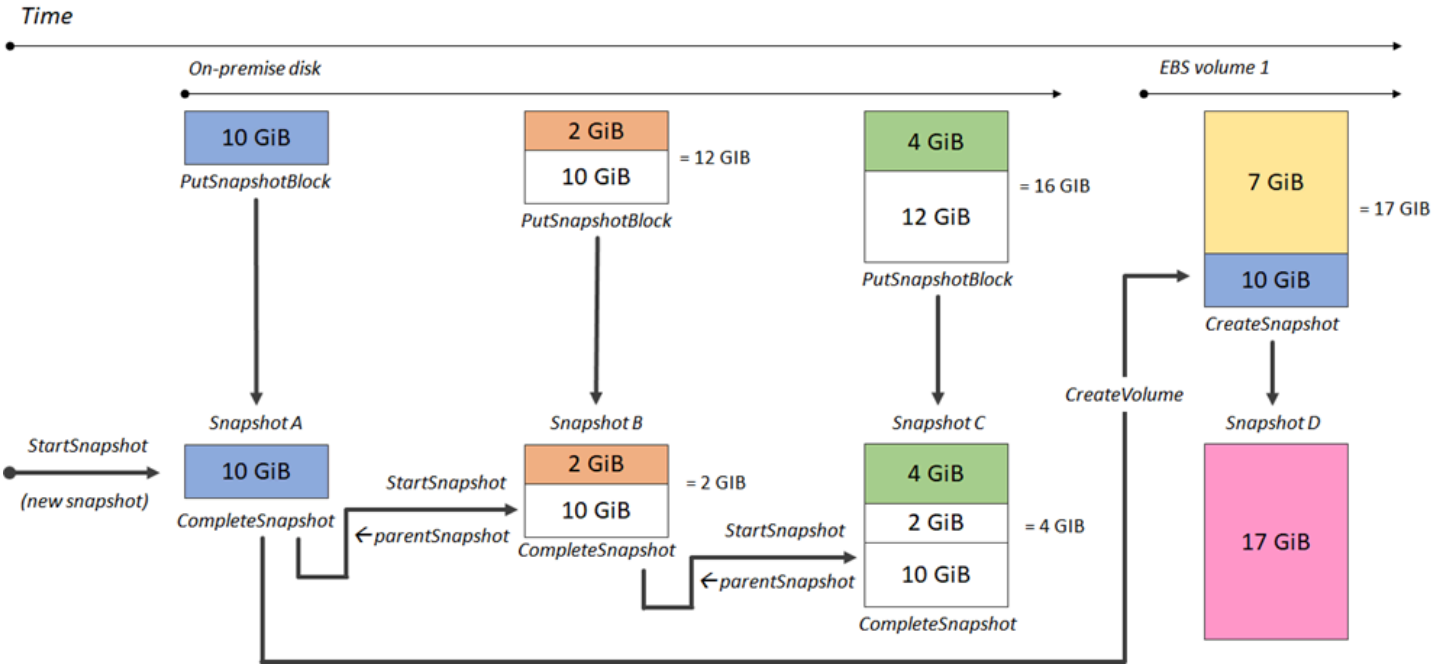
Tulis snapshot Amazon EBS dengan API langsung EBS

Langkah-langkah berikut menjelaskan cara menggunakan API langsung EBS untuk menulis snapshot inkremental:

1. Gunakan StartSnapshot tindakan dan tentukan ID snapshot induk untuk memulai snapshot sebagai snapshot tambahan dari yang sudah ada, atau hilangkan ID snapshot induk untuk memulai snapshot baru. Tindakan ini mengembalikan ID snapshot baru, yang berada dalam status tertunda.
2. Gunakan PutSnapshotBlock tindakan dan tentukan ID snapshot yang tertunda untuk menambahkan data ke dalamnya dalam bentuk blok individual. Anda harus menentukan checksum Base64-encoded SHA256 untuk blok data yang dikirimkan. Layanan ini menghitung checksum data yang diterima dan memvalidasinya dengan checksum yang Anda tentukan. Tindakan gagal jika checksum tidak cocok.
3. Setelah selesai menambahkan data ke snapshot yang tertunda, gunakan tindakan CompleteSnapshot untuk memulai alur kerja asinkronous yang menyegel snapshot dan mengubah statusnya menjadi selesai.

Ulangi langkah-langkah ini untuk membuat snapshot inkremental baru menggunakan snapshot yang dibuat sebelumnya sebagai induk.

Misalnya, dalam diagram berikut, snapshot A adalah snapshot baru pertama yang dimulai. Snapshot A digunakan sebagai snapshot induk untuk memulai snapshot B. Snapshot B digunakan sebagai snapshot induk untuk memulai dan membuat snapshot C. Snapshot A, B, dan C adalah snapshot inkremental. Snapshot A digunakan untuk membuat volume EBS 1. Snapshot D dibuat dari volume EBS 1. Snapshot D adalah snapshot inkremental A; bukan snapshot inkremental dari B atau C.



Contoh berikut menunjukkan cara menulis snapshot menggunakan API langsung EBS.

Topik

- Mulai snapshot
- Menempatkan data ke dalam snapshot
- Menyelesaikan snapshot

Mulai snapshot

AWS CLI

Contoh perintah start-snapshot berikut memulai snapshot 8 GiB, menggunakan snapshot `snap-123EXAMPLE1234567` sebagai snapshot induk. Snapshot baru akan berupa snapshot inkremental dari snapshot induk. Snapshot berpindah ke status kesalahan jika tidak ada permintaan put atau complete yang dibuat untuk snapshot dalam periode waktu tunggu `60` menit yang ditentukan. Token klien `550e8400-e29b-41d4-a716-446655440000` memastikan idempotensi permintaan tersebut. Jika token klien dihilangkan, AWS SDK secara otomatis menghasilkannya untuk Anda. Untuk informasi selengkapnya tentang idempotensi, lihat Pastikan idempotensi dalam permintaan API StartSnapshot.

```
aws ebs start-snapshot --volume-size 8 --parent-snapshot snap-123EXAMPLE1234567 --
timeout 60 --client-token 550e8400-e29b-41d4-a716-446655440000
```

Contoh respons berikut untuk perintah sebelumnya menunjukkan ID snapshot, ID akun AWS , status, ukuran volume dalam GiB, dan ukuran blok di snapshot. Snapshot dimulai dalam status `pending`. Tentukan ID snapshot di bagian perintah `put-snapshot-block` berikutnya untuk menuliskan data ke snapshot, lalu menggunakan perintah `complete-snapshot` untuk menyelesaikan snapshot dan mengubahnya status menjadi `completed`.

```
{
    "SnapshotId": "snap-0aaEXAMPLEe306d62",
    "OwnerId": "111122223333",
    "Status": "pending",
    "VolumeSize": 8,
    "BlockSize": 524288
}
```

AWS API

StartSnapshotContoh permintaan berikut memulai snapshot 8 GiB, menggunakan snapshot `snap-123EXAMPLE1234567` sebagai snapshot induk. Snapshot baru akan berupa snapshot inkremental dari snapshot induk. Snapshot berpindah ke status kesalahan jika tidak ada permintaan put atau complete yang dibuat untuk snapshot dalam periode waktu tunggu `60` menit yang ditentukan. Token klien `550e8400-e29b-41d4-a716-446655440000` memastikan idempotensi permintaan tersebut. Jika token klien dihilangkan, AWS SDK secara otomatis menghasilkannya untuk Anda. Untuk informasi selengkapnya tentang idempotensi, lihat Pastikan idempotensi dalam permintaan API StartSnapshot.

```
POST /snapshots HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T040724Z
Authorization: <Authentication parameter>

{
    "VolumeSize": 8,
    "ParentSnapshot": snap-123EXAMPLE1234567,
    "ClientToken": "550e8400-e29b-41d4-a716-446655440000",
    "Timeout": 60
}
```

Contoh respons berikut untuk permintaan sebelumnya menunjukkan ID snapshot, ID akun AWS , status, ukuran volume dalam GiB, dan ukuran blok di snapshot. Snapshot dimulai dalam status tertunda. Tentukan ID snapshot di permintaan `PutSnapshotBlocks` berikutnya untuk menuliskan data ke snapshot.

```
HTTP/1.1 201 Created
x-amzn-RequestId: 929e6eb9-7183-405a-9502-5b7da37c1b18
Content-Type: application/json
Content-Length: 181
Date: Thu, 18 Jun 2020 04:07:29 GMT
Connection: keep-alive

{
    "BlockSize": 524288,
    "Description": null,
    "OwnerId": "138695307491",
```

```
    "Progress": null,
    "SnapshotId": "snap-052EXAMPLEc85d8dd",
    "StartTime": null,
    "Status": "pending",
    "Tags": null,
    "VolumeSize": 8
}
```

Menempatkan data ke dalam snapshot

AWS CLI

Perintah contoh put-snapshot-block berikut menulis 524288 Bytes data untuk memblokir indeks pada snapshot. 1000 snap-0aaEXAMPLEe306d62 Checksum Q0D3gmEQ0XATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM= berkode Base64 dibuat menggunakan algoritme SHA256. Data yang ditransmisikan terdapat di file /tmp/data.

```
aws ebs put-snapshot-block --snapshot-id snap-0aaEXAMPLEe306d62
 --block-index 1000 --data-length 524288 --block-data /tmp/data --
checksum Q0D3gmEQ0XATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM= --checksum-algorithm SHA256
```

Contoh respons untuk perintah sebelumnya berikut ini mengonfirmasi panjang data, checksum, dan algoritma checksum untuk data yang diterima oleh layanan.

```
{
    "DataLength": "524288",
    "Checksum": "Q0D3gmEQ0XATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=",
    "ChecksumAlgorithm": "SHA256"
}
```

AWS API

PutSnapshotContoh permintaan berikut menulis 524288 Bytes data untuk memblokir indeks 1000 pada snapshotsnap-052EXAMPLEc85d8dd. Checksum Q0D3gmEQ0XATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM= dengan enkode Base64 dibuat menggunakan algoritma SHA256. Data ditransmisikan dalam badan permintaan dan direpresentasikan seperti *BlockData* dalam contoh berikut.

```
PUT /snapshots/snap-052EXAMPLEc85d8dd/blocks/1000 HTTP/1.1
```

```
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
x-amz-Data-Length: 524288
x-amz-Checksum: QOD3gmEQOXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=
x-amz-Checksum-Algorithm: SHA256
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T042215Z
X-Amz-Content-SHA256: UNSIGNED-PAYLOAD
Authorization: <Authentication parameter>

          BlockData
```

Contoh respons untuk permintaan sebelumnya berikut ini mengonfirmasi panjang data, checksum, dan algoritma checksum untuk data yang diterima oleh layanan.

```
HTTP/1.1 201 Created
x-amzn-RequestId: 643ac797-7e0c-4ad0-8417-97b77b43c57b
x-amz-Checksum: QOD3gmEQOXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=
x-amz-Checksum-Algorithm: SHA256
Content-Type: application/json
Content-Length: 2
Date: Thu, 18 Jun 2020 04:22:12 GMT
Connection: keep-alive

{}
```

Menyelesaikan snapshot

AWS CLI

Contoh perintah complete-snapshot berikut menyelesaikan snapshot `snap-0aaEXAMPLEe306d62`. Perintah menentukan bahwa 5 blok ditulis untuk snapshot. Checksum `6D3nmwi5f2F0wlh7xX8QprrJBFzDX8aacdOcA3KCM3c=` mewakili checksum untuk set data lengkap yang ditulis ke snapshot. Untuk informasi selengkapnya tentang checksum, lihat Gunakan checksum API langsung EBS untuk memvalidasi data snapshot sebelumnya di dalam panduan ini.

```
aws ebs complete-snapshot --snapshot-id snap-0aaEXAMPLEe306d62 --changed-blocks-
count 5 --checksum 6D3nmwi5f2F0wlh7xX8QprrJBFzDX8aacdOcA3KCM3c= --checksum-
algorithm SHA256 --checksum-aggregation-method LINEAR
```

Berikut ini adalah contoh tanggapan untuk perintah sebelumnya.

```
{
    "Status": "pending"
}
```

AWS API

CompleteSnapshotContoh permintaan berikut melengkapi snapshot`snap-052EXAMPLEc85d8dd`. Perintah menentukan bahwa 5 blok ditulis untuk snapshot. Checksum `6D3nmwi5f2F0wlh7xX8QprrJBFzDX8aacdOcA3KCM3c=` merepresentasikan checksum untuk set data lengkap yang ditulis ke snapshot.

```
POST /snapshots/completion/snap-052EXAMPLEc85d8dd HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
x-amz-ChangedBlocksCount: 5
x-amz-Checksum: 6D3nmwi5f2F0wlh7xX8QprrJBFzDX8aacdOcA3KCM3c=
x-amz-Checksum-Algorithm: SHA256
x-amz-Checksum-Aggregation-Method: LINEAR
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T043158Z
Authorization: <Authentication parameter>
```

Berikut ini adalah contoh tanggapan untuk permintaan sebelumnya.

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: 06cba5b5-b731-49de-af40-80333ac3a117
Content-Type: application/json
Content-Length: 20
Date: Thu, 18 Jun 2020 04:31:50 GMT
Connection: keep-alive

{"Status":"pending"}
```

Hasil enkripsi untuk API langsung EBS

Saat Anda memulai snapshot baru menggunakan StartSnapshot, status enkripsi tergantung pada nilai yang Anda tentukan untuk Encry pted, KmsKeyArn, dan ParentSnapshotId, dan apakah AWS akun Anda diaktifkan untuk enkripsi secara default.

> **Note**
>
> - Anda mungkin memerlukan izin IAM tambahan untuk menggunakan API langsung EBS dengan enkripsi. Untuk informasi selengkapnya, lihat Izin untuk menggunakan AWS KMS keys.
> - Jika enkripsi Amazon EBS secara default diaktifkan di AWS akun Anda, Anda tidak dapat membuat snapshot yang tidak dienkripsi.
> - Jika enkripsi Amazon EBS secara default diaktifkan di AWS akun Anda, Anda tidak dapat memulai snapshot baru menggunakan snapshot induk yang tidak dienkripsi. Anda harus terlebih dahulu mengenkripsi snapshot induk dengan menyalinnya. Untuk informasi selengkapnya, lihat Menyalin snapshot Amazon EBS.

Topik

- Hasil enkripsi: Snapshot induk yang tidak terenkripsi
- Hasil enkripsi: Snapshot induk yang tidak terenkripsi
- Hasil enkripsi: Tidak ada snapshot induk

Hasil enkripsi: Snapshot induk yang tidak terenkripsi

Tabel berikut menjelaskan hasil enkripsi untuk setiap kombinasi pengaturan yang memungkinkan.

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
Tidak terenkripsi	Diabaikan	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	Snapshot tidak terenkripsi.
		Ditentukan	Aktif	
			Nonaktif	

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
Tidak terenkripsi	Benar	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	
		Ditentukan	Aktif	
			Nonaktif	
Tidak terenkripsi	Salah	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	
		Ditentukan	Aktif	
			Nonaktif	

Hasil enkripsi: Snapshot induk yang tidak terenkripsi

Tabel berikut menjelaskan hasil enkripsi untuk setiap kombinasi pengaturan yang memungkinkan.

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
Dienkripsi	Diabaikan	Diabaikan	Aktif	Snapshot dienkripsi menggunakan kunci KMS yang sama dengan snapshot induk.
			Nonaktif	
		Ditentukan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	
Dienkripsi	Benar	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
			Nonaktif	
		Ditentukan	Aktif	
			Nonaktif	
Dienkripsi	Salah	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	
		Ditentukan	Aktif	
			Nonaktif	

Hasil enkripsi: Tidak ada snapshot induk

Tabel berikut menjelaskan hasil enkripsi untuk setiap kombinasi pengaturan yang memungkinkan.

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
Diabaikan	Benar	Diabaikan	Aktif	Snapshot dienkripsi menggunakan kunci KMS default untuk akun Anda. *
			Nonaktif	
		Ditentukan	Diaktifkan	Snapshot dienkripsi menggunakan kunci KMS yang ditentukan untuk KmsKeyArn.
			Nonaktif	
Diabaikan	Salah	Diabaikan	Aktif	Permintaan gagal dengan `ValidationException` .

ParentSnapshotId	Dienkripsi	KmsKeyArn	Enkripsi secara default	Hasil
			Nonaktif	Snapshot tidak terenkripsi.
		Ditentukan	Aktif	Permintaan gagal dengan `ValidationException` .
			Nonaktif	
Diabaikan	Diabaikan	Diabaikan	Aktif	Snapshot dienkripsi menggunakan kunci KMS default untuk akun Anda. *
			Nonaktif	Snapshot tidak terenkripsi.
		Ditentukan	Diaktifkan	Snapshot dienkripsi menggunakan kunci KMS yang ditentukan untuk KmsKeyArn.
			Nonaktif	

* Kunci KMS default ini bisa berupa kunci yang dikelola pelanggan atau kunci KMS terkel AWS ola default untuk enkripsi Amazon EBS.

Gunakan checksum API langsung EBS untuk memvalidasi data snapshot

T GetSnapshotBlock indakan mengembalikan data yang berada dalam blok snapshot, dan PutSnapshotBlock tindakan menambahkan data ke blok dalam snapshot. Data blok yang tidak ditransmisikan sebagai bagian dari proses penandatanganan Signature Versi 4. Oleh karena itu, checksum digunakan untuk memvalidasi integritas data sebagai berikut:

- Saat Anda menggunakan GetSnapshotBlock tindakan, respons memberikan checksum Base64-encoded SHA256 untuk data blok menggunakan header x-AMZ-checksum, dan algoritma checksum menggunakan header x-amz-. Checksum-Algorithm Gunakan checksum yang dikembalikan untuk memvalidasi integritas data. Jika checksum yang Anda hasilkan tidak sesuai

dengan yang diberikan oleh Amazon EBS, Anda harus mempertimbangkan data yang tidak valid dan mencoba kembali permintaan Anda.

- Saat Anda menggunakan PutSnapshotBlock tindakan, permintaan Anda harus memberikan checksum Base64-encoded SHA256 untuk data blok menggunakan header x-AMZ-checksum, dan algoritma checksum menggunakan header x-amz-. Checksum-Algorithm Checksum yang Anda berikan divalidasi dengan checksum yang dibuat oleh Amazon EBS untuk memvalidasi integritas data. Jika checksum tidak sesuai, permintaan gagal.
- Saat Anda menggunakan CompleteSnapshot tindakan, permintaan Anda dapat secara opsional memberikan checksum Base64-encoded SHA256 agregat untuk kumpulan data lengkap yang ditambahkan ke snapshot. Berikan checksum menggunakan header x-AMZ-checksum, algoritma checksum menggunakan header x-amz-, dan metode agregasi checksum menggunakan Checksum-Algorithm header x-amz-. Checksum-Aggregation-Method Untuk membuat checksum gabungan menggunakan metode agregasi linear, mengatur checksum untuk setiap blok dalam urutan naik indeks blok mereka, menyatukan mereka untuk membentuk satu string, kemudian membuat checksum pada seluruh rangkaian menggunakan SHA256.//

Checksum dalam tindakan ini merupakan bagian dari proses penandatanganan Signature Versi 4.

Pastikan idempotensi dalam permintaan API StartSnapshot

Idempotensi memastikan bahwa permintaan API hanya selesai satu kali. Dengan permintaan idempotensi, jika permintaan asli selesai, percobaan berikutnya mengembalikan hasil dari permintaan awal yang berhasil dan tidak memiliki efek tambahan.

API StartSnapshot mendukung idempotensi menggunakan token klien. Token klien adalah string unik yang Anda tentukan saat membuat permintaan API. Jika Anda mencoba ulang permintaan API dengan token klien yang sama dan parameter permintaan yang sama setelah berhasil diselesaikan, hasil permintaan awal akan dikembalikan. Jika Anda mencoba ulang permintaan dengan token klien yang sama, tetapi mengubah satu atau beberapa parameter permintaan, kesalahan `ConflictException` dikembalikan.

Jika Anda tidak menentukan token klien Anda sendiri, AWS SDK secara otomatis menghasilkan token klien untuk permintaan untuk memastikan bahwa token itu idempotent.

Token klien dapat berupa string yang mencakup hingga 64 karakter ASCII. Anda tidak boleh menggunakan kembali token klien yang sama untuk permintaan yang berbeda.

Untuk membuat StartSnapshot permintaan idempotent dengan token klien Anda sendiri menggunakan API

Tentukan parameter permintaan `ClientToken`.

```
POST /snapshots HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T040724Z
Authorization: <Authentication parameter>

{
    "VolumeSize": 8,
    "ParentSnapshot": snap-123EXAMPLE1234567,
    "ClientToken": "550e8400-e29b-41d4-a716-446655440000",
    "Timeout": 60
}
```

Untuk membuat StartSnapshot permintaan idempotent dengan token klien Anda sendiri menggunakan AWS CLI

Tentukan parameter permintaan `client-token`.

```
$ C:\> aws ebs start-snapshot --region us-east-2 --volume-size 8 --parent-
snapshot snap-123EXAMPLE1234567 --timeout 60 --client-token 550e8400-e29b-41d4-
a716-446655440000
```

Percobaan ulang kesalahan untuk API langsung EBS

AWS SDK menerapkan logika coba ulang otomatis untuk permintaan yang mengembalikan respons kesalahan. Anda dapat mengonfigurasi pengaturan coba lagi untuk AWS SDK. Untuk informasi selengkapnya, lihat dokumentasi SDK Anda.

Anda dapat mengonfigurasi AWS CLI agar secara otomatis mencoba kembali beberapa permintaan yang gagal. Untuk informasi selengkapnya tentang mengonfigurasi percobaan ulang untuk AWS CLI, lihat AWS CLI percobaan ulang di Panduan AWS Command Line Interface Pengguna.

API Kueri AWS tidak mendukung logika coba ulang untuk permintaan yang gagal. Jika Anda menggunakan permintaan HTTP atau HTTPS, Anda harus menerapkan logika coba lagi dalam aplikasi klien Anda.

Tabel berikut menunjukkan kemungkinan respons kesalahan API. Beberapa kesalahan API dapat dicoba ulang. Aplikasi klien Anda harus selalu mencoba ulang permintaan gagal yang menerima kesalahan yang dapat dicoba ulang.

Kesalahan	Kode respons	Deskripsi	Dilempar oleh	Dicoba ulang?
`InternalServerException`	500	Permintaan gagal karena masalah jaringan atau AWS sisi server.	Semua API	Ya
`ThrottlingException`	400	Jumlah permintaan API telah melampaui batas throttling permintaan API maksimum yang diizinkan untuk akun.	Semua API	Ya
`RequestThrottleException`	400	Jumlah permintaan API telah melampaui batas throttling permintaan API maksimum yang diizinkan untuk snapshot.	GetSnapshotBlock | PutSnapshotBlock	Ya
`ValidationException` dengan pesan `"Failed to read block data"`	400	Blok data yang disediakan tidak dapat dibaca.	PutSnapshotBlock	Ya

Kesalahan	Kode respons	Deskripsi	Dilempar oleh	Dicoba ulang?
`ValidationException` dengan pesan lainnya	400	Sintaksis permintaan salah format, atau input tidak memenuhi batasan yang ditentukan oleh Layanan AWS.	Semua API	Tidak
`ResourceNotFoundException`	404	ID snapshot yang ditentukan tidak ada.	Semua API	Tidak
`ConflictException`	409	Token klien yang ditentukan sebelumnya digunakan dalam permintaan serupa yang memiliki parameter permintaan berbeda. Untuk informasi selengkapnya, lihat Pastikan idempotensi dalam permintaan API StartSnapshot.	StartSnapshot	Tidak

Kesalahan	Kode respons	Deskripsi	Dilempar oleh	Dicoba ulang?
`AccessDeniedException`	403	Anda tidak memiliki izin untuk melakukan operasi yang diminta.	Semua API	Tidak
`ServiceQuotaExceededException`	402	Permintaan gagal karena memenuhi permintaan akan melebihi satu atau lebih kuota layanan dependen untuk akun Anda.	Semua API	Tidak
`InvalidSignatureException`	403	Tanda tangan otorisasi permintaan telah kedaluwarsa. Anda dapat mencoba lagi permintaan hanya setelah menyegarkan tanda tangan otorisasi.	Semua API	Tidak

Mengoptimalkan kinerja untuk API langsung EBS

Anda dapat menjalankan permintaan API secara bersamaan. Dengan asumsi PutSnapshotBlock latensi 100ms, maka thread dapat memproses 10 permintaan dalam satu detik. Selain itu, dengan asumsi aplikasi klien Anda menciptakan beberapa alur dan koneksi (misalnya, 100 koneksi),

dapat membuat 1000 permintaan (10 * 100) per detik secara keseluruhan. Ini akan sesuai dengan throughput sekitar 500 MB per detik.

Daftar berikut ini berisi beberapa hal yang harus dicari dalam aplikasi Anda:

- Apakah setiap utas menggunakan koneksi terpisah? Jika koneksi dibatasi pada aplikasi, maka banyak alur akan menunggu koneksi tersedia dan Anda akan melihat throughput yang lebih rendah.
- Apakah ada waktu tunggu dalam aplikasi di antara dua permintaan yang dimasukkan? Hal ini akan mengurangi throughput alur yang efektif.
- Batas bandwidth pada instans - Jika bandwidth pada instasn dibagikan oleh aplikasi lain, maka dapat membatasi keluaran yang tersedia untuk permintaan PutSnapshotBlock.

Pastikan untuk mencatat beban kerja lain yang mungkin berjalan di akun untuk menghindari hambatan. Anda juga harus membangun mekanisme coba ulang ke dalam alur kerja API langsung EBS untuk menangani throttling, waktu habis, dan ketidaktersediaan layanan.

Tinjau kuota layanan API langsung EBS untuk menentukan permintaan API maksimum yang dapat Anda jalankan per detik. Untuk informasi selengkapnya, lihat Titik Akhir dan Kuota Amazon Elastic Block Store dalam Referensi Umum AWS .

Titik akhir layanan untuk API langsung EBS

Endpoint adalah URL yang berfungsi sebagai titik masuk untuk layanan AWS web. API langsung EBS mendukung jenis titik akhir berikut:

- Titik akhir IPv4
- Dual-stack Endpoint yang mendukung IPv4 dan IPv6
- Titik akhir FIPS

Saat Anda membuat permintaan, Anda dapat menentukan titik akhir dan Wilayah yang akan digunakan. Jika Anda tidak menentukan titik akhir, titik akhir IPv4 digunakan secara default. Untuk menggunakan tipe titik akhir yang berbeda, Anda harus menentukannya dalam permintaan Anda. Untuk contoh cara melakukannya, lihat Menentukan titik akhir.

Untuk informasi selengkapnya tentang Wilayah, lihat Wil ayah dan Zona Ketersediaan di Panduan Pengguna Amazon EC2. Untuk daftar titik akhir untuk API langsung EBS, lihat Titik akhir untuk API langsung EBS di Referensi Umum Amazon Web Services.

Topik

- Titik akhir IPv4
- Dual-stack Titik akhir (IPv4 dan IPv6)
- Titik akhir FIPS
- Menentukan titik akhir

Titik akhir IPv4

Titik akhir IPv4 hanya mendukung lalu lintas IPv4. Titik akhir IPv4 tersedia untuk semua Wilayah.

API langsung EBS hanya mendukung titik akhir IPv4 Regional yang dapat Anda gunakan untuk membuat permintaan Anda. Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Nama titik akhir menggunakan konvensi penamaan berikut:

- `ebs.region.amazonaws.com`

Misalnya, untuk mengarahkan permintaan Anda ke titik akhir `us-east-2` IPv4, Anda harus menentukan `ebs.us-east-2.amazonaws.com` sebagai titik akhir. Untuk daftar titik akhir untuk API langsung EBS, lihat Titik akhir untuk API langsung EBS di Referensi Umum Amazon Web Services.

Harga

Anda tidak dikenai biaya untuk data yang ditransfer langsung antara API langsung EBS dan instans Amazon EC2 menggunakan titik akhir IPv4 di Wilayah yang sama. Namun, jika ada layanan perantara, seperti AWS PrivateLink titik akhir, NAT Gateway, atau Amazon VPC Transit Gateway, Anda akan dikenakan biaya terkait.

Dual-stack Titik akhir (IPv4 dan IPv6)

Dual-stack Endpoint mendukung lalu lintas IPv4 dan IPv6. Dual-stack titik akhir tersedia untuk semua Wilayah.

Untuk menggunakan IPv6, Anda harus menggunakan titik akhir tumpukan ganda. Saat Anda membuat permintaan ke titik akhir tumpukan ganda, URL titik akhir memutuskan ke alamat IPv6 atau IPv4, tergantung pada protokol yang digunakan oleh jaringan dan klien Anda.

API langsung EBS hanya mendukung titik akhir dual-stack regional, yang berarti Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Dual-stack nama titik akhir menggunakan konvensi penamaan berikut:

- `ebs.region.api.aws`

Misalnya, nama titik akhir tumpukan ganda untuk Wilayah `eu-west-1` adalah `ebs.eu-west-1.api.aws`. Untuk daftar titik akhir untuk API langsung EBS, lihat Titik akhir untuk API langsung EBS di Referensi Umum Amazon Web Services.

Harga

Anda tidak dikenai biaya untuk data yang ditransfer langsung antara API langsung EBS dan instans Amazon EC2 menggunakan titik akhir tumpukan ganda di Wilayah yang sama. Namun, jika ada layanan perantara, seperti AWS PrivateLink titik akhir, NAT Gateway, atau Amazon VPC Transit Gateway, Anda akan dikenakan biaya terkait.

Titik akhir FIPS

API langsung EBS menyediakan titik akhir FIPS-validated IPv4 dan dual-stack (IPv4 dan IPv6) untuk Wilayah berikut:

- `us-east-1` — AS Timur (Virginia Utara)
- `us-east-2` — AS Timur (Ohio)
- `us-west-1` — AS Barat (California Utara)
- `us-west-2` — AS Barat (Oregon)
- `ca-central-1` – Kanada (Pusat)
- `ca-west-1`— Kanada Barat (Calgary)

Titik akhir IPv4 FIPS menggunakan konvensi penamaan berikut: `ebs-fips.region.amazonaws.com`. Misalnya, titik akhir IPv4 FIPS untuk `us-east-1` adalah `ebs-fips.us-east-1.amazonaws.com`.

Titik akhir tumpukan ganda FOPS menggunakan konvensi penamaan berikut: `ebs-fips.region.api.aws`. Misalnya, titik akhir tumpukan ganda FIPS untuk `us-east-1` adalah `ebs-fips.us-east-1.api.aws`.

Untuk informasi selengkapnya tentang titik akhir FIPS, lihat Titik akhir FIPS di Referensi Umum Amazon Web Services.

Menentukan titik akhir

Bagian ini memberikan beberapa contoh cara menentukan titik akhir saat membuat permintaan.

AWS CLI

Contoh berikut menunjukkan cara menentukan titik akhir untuk Wilayah `us-east-2` menggunakan AWS CLI.

- Dual-stack

```
aws ebs list-snapshot-blocks --snapshot-id snap-0987654321 --starting-block-index
 1000 --endpoint-url https://ebs.us-east-2.api.aws
```

- IPv4

```
aws ebs list-snapshot-blocks --snapshot-id snap-0987654321 --starting-block-index
 1000 --endpoint-url https://ebs.us-east-2.amazonaws.com
```

AWS SDK for Java 2.x

Contoh berikut menunjukkan cara menentukan titik akhir untuk Wilayah `us-east-2` menggunakan AWS SDK for Java 2.x.

- Dual-stack

```
AwsClientBuilder.EndpointConfiguration config = new
 AwsClientBuilder.EndpointConfiguration("https://ebs.us-east-2.api.aws", "us-
east-2");
AmazonEBS ebs = AmazonEBSClientBuilder.standard()
    .withEndpointConfiguration(config)
    .build();
```

- IPv4

```
AwsClientBuilder.EndpointConfiguration config = new
 AwsClientBuilder.EndpointConfiguration("https://ebs.us-east-2.amazonaws.com",
 "us-east-2");
AmazonEBS ebs = AmazonEBSClientBuilder.standard()
    .withEndpointConfiguration(config)
    .build();
```

AWS SDK for Go

Contoh berikut menunjukkan cara menentukan titik akhir untuk Wilayah `us-east-2` menggunakan AWS SDK untuk Go.

- Dual-stack

```
sess := session.Must(session.NewSession())
svc := ebs.New(sess, &aws.Config{
    Region: aws.String(endpoints.UsEast2RegionID),
    Endpoint: aws.String("https://ebs.us-east-2.api.aws")
})
```

- IPv4

```
sess := session.Must(session.NewSession())
svc := ebs.New(sess, &aws.Config{
    Region: aws.String(endpoints.UsEast2RegionID),
    Endpoint: aws.String("https://ebs.us-east-2.amazonaws.com")
})
```

AWS Contoh kode SDK untuk API langsung EBS

Contoh kode berikut menunjukkan cara menggunakan API langsung EBS dengan AWS perangkat pengembangan perangkat lunak (SDK).

Tindakan

- Gunakan StartSnapshot dengan AWS SDK atau CLI
- Gunakan PutSnapshotBlock dengan AWS SDK atau CLI
- Gunakan CompleteSnapshot dengan AWS SDK atau CLI

Gunakan **`StartSnapshot`** dengan AWS SDK atau CLI

Contoh kode berikut menunjukkan cara menggunakan`StartSnapshot`.

Rust

SDK for Rust

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
async fn start(client: &Client, description: &str) -> Result<String, Error> {
    let snapshot = client
        .start_snapshot()
        .description(description)
        .encrypted(false)
        .volume_size(1)
        .send()
        .await?;

    Ok(snapshot.snapshot_id.unwrap())
}
```

- Untuk detail API, lihat StartSnapshot di AWS SDK untuk referensi API Rust.

Gunakan **`PutSnapshotBlock`** dengan AWS SDK atau CLI

Contoh kode berikut menunjukkan cara menggunakan`PutSnapshotBlock`.

Rust

SDK for Rust

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
async fn add_block(
    client: &Client,
    id: &str,
    idx: usize,
    block: Vec<u8>,
    checksum: &str,
) -> Result<(), Error> {
    client
        .put_snapshot_block()
        .snapshot_id(id)
        .block_index(idx as i32)
        .block_data(ByteStream::from(block))
        .checksum(checksum)
        .checksum_algorithm(ChecksumAlgorithm::ChecksumAlgorithmSha256)
        .data_length(EBS_BLOCK_SIZE as i32)
        .send()
        .await?;

    Ok(())
}
```

- Untuk detail API, lihat PutSnapshotBlock di AWS SDK untuk referensi API Rust.

Gunakan **`CompleteSnapshot`** dengan AWS SDK atau CLI

Contoh kode berikut menunjukkan cara menggunakan`CompleteSnapshot`.

Rust

SDK for Rust

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
async fn finish(client: &Client, id: &str) -> Result<(), Error> {
    client
        .complete_snapshot()
        .changed_blocks_count(2)
        .snapshot_id(id)
        .send()
        .await?;

    println!("Snapshot ID {}", id);
    println!("The state is 'completed' when all of the modified blocks have been
 transferred to Amazon S3.");
    println!("Use the get-snapshot-state code example to get the state of the
 snapshot.");

    Ok(())
}
```

- Untuk detail API, lihat CompleteSnapshot di AWS SDK untuk referensi API Rust.

Membuat koneksi pribadi antara VPC dan API langsung EBS

Anda dapat membangun koneksi privat antara VPC dan API langsung EBS dengan membuat titik akhir VPC antarmuka, yang didukung oleh AWS PrivateLink. Anda dapat mengakses API langsung EBS 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 berkomunikasi dengan API langsung EBS.

Kami membuat antarmuka jaringan endpoint di setiap subnet yang Anda aktifkan untuk titik akhir antarmuka.

Untuk informasi selengkapnya, lihat Ak AWS PrivateLink ses Layanan AWS melalui dalam AWS PrivateLink Panduan.

Pertimbangan untuk titik akhir VPC API langsung EBS

Sebelum Anda menyiapkan titik akhir VPC antarmuka untuk API langsung EBS, tinjau Pertimbangan di Panduan AWS PrivateLink .

Secara default, akses penuh ke API langsung EBS diizinkan melalui titik akhir. Anda dapat mengontrol akses ke titik akhir antarmuka menggunakan kebijakan titik akhir VPC. Anda dapat melampirkan kebijakan titik akhir ke titik akhir VPC yang mengontrol akses ke API langsung EBS. Kebijakan titik akhir menentukan informasi berikut:

- Kepala sekolah yang dapat melakukan tindakan.
- T ind akan yang bisa dilakukan.
- Sumber daya di mana tindakan dapat dilakukan.

Untuk informasi selengkapnya, lihat Mengendalikan Akses ke Layanan dengan Titik Akhir VPC dalam Panduan Pengguna VPC Amazon.

Berikut ini adalah contoh kebijakan titik akhir untuk API langsung EBS. Saat dilampirkan ke titik akhir, kebijakan ini memberikan akses ke semua tindakan API langsung EBS pada semua sumber daya, kecuali snapshot yang diberi tag dengan kunci `Environment` dan nilai. `Test`

```
{
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ebs:*",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Environment": "Test"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ebs:*",
```

```
            "Principal": "*",
            "Resource": "*"
        }
    ]
}
```

Buat suatu titik akhir VPC antarmuka untuk API langsung EBS

Anda dapat membuat titik akhir VPC untuk API langsung EBS menggunakan konsol Amazon VPC atau AWS Command Line Interface (AWS CLI). Untuk informasi selengkapnya, lihat Membuat titik akhir VPC di Panduan Pengguna AWS PrivateLink .

Buat titik akhir VPC untuk API langsung EBS menggunakan salah satu nama layanan berikut:

- `com.amazonaws.region.ebs`
- `com.amazonaws.region.ebs-fips`— Untuk membuat titik akhir VPC antarmuka yang sesuai dengan Standar Pemrosesan Informasi Federal (FIPS) Publikasi 140-2 standar pemerintah AS.

> **Note**
>
> FIPS-compliant antarmuka titik akhir VPC dapat dibuat untuk Wilayah berikut: `us-east-1` | | | `us-east-2` | `us-west-1` `us-west-2` |`ca-central-1`. `ca-west-1` FIPS-compliant antarmuka titik akhir VPC mendukung lalu lintas IPv4 dan IPv6.

Jika Anda mengaktifkan DNS privat untuk titik akhir, Anda dapat mengajukan permintaan API ke API langsung EBS menggunakan nama DNS default untuk Wilayah, misalnya, `ebs.us-east-1.amazonaws.com`.

Log panggilan API langsung EBS menggunakan AWS CloudTrail

API langsung EBS terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan. CloudTrail menangkap panggilan yang dilakukan ke API langsung EBS sebagai peristiwa. Panggilan yang ditangkap termasuk panggilan dari Konsol Manajemen AWS dan panggilan kode ke API langsung EBS. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke API langsung EBS, alamat IP dari mana permintaan dibuat, kapan permintaan dibuat, dan detail tambahan.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut hal ini:

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna.
- Apakah permintaan dibuat atas nama pengguna Pusat Identitas IAM.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

CloudTrail aktif di akun Anda Akun AWS saat Anda membuat akun dan Anda secara otomatis memiliki akses ke riwayat CloudTrail Acara. Ri CloudTrail wayat acara menyediakan catatan yang dapat dilihat, dapat dicari, dapat diunduh, dan tidak dapat diubah dari 90 hari terakhir dari peristiwa manajemen yang direkam dalam sebuah. Wilayah AWS Untuk informasi selengkapnya, lihat Bek erja dengan riwayat CloudTrail acara di Panduan AWS CloudTrail Pengguna. Tidak ada CloudTrail biaya untuk melihat ri wayat acara.

Untuk catatan peristiwa yang berkelanjutan dalam 90 hari Akun AWS terakhir Anda, buat jejak atau penyimpanan data acara CloudTrail Lake.

CloudTrail jalan setapak

Jej ak CloudTrail memungkinkan pengiriman file log ke bucket Amazon S3. Semua jalur yang dibuat menggunakan Multi-wilayah. Konsol Manajemen AWS Anda dapat membuat jejak wilayah tunggal atau multi-wilayah dengan menggunakan. AWS CLI Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas di semua Wilayah AWS di akun Anda. Jika Anda membuat jejak satu wilayah, Anda hanya dapat melihat peristiwa yang dicatat di jalur tersebut Wilayah AWS. Untuk informasi selengkapnya tentang jalur, lihat Membuat jejak untuk Anda Akun AWS dan Membuat jejak untuk organisasi di Panduan AWS CloudTrail Pengguna.

Anda dapat mengirimkan satu salinan acara manajemen yang sedang berlangsung ke bucket Amazon S3 Anda tanpa biaya CloudTrail dengan membuat jejak, namun, ada biaya penyimpanan Amazon S3. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga. Untuk informasi tentang harga Amazon S3, lihat Harga Amazon S3.

CloudTrail Penyimpanan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan SQL-based kueri pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam format JSON berbasis baris ke format

Apache ORC. ORC adalah format penyimpanan kolumnar yang dioptimalkan untuk pengambilan data dengan cepat. Peristiwa digabungkan ke dalam penyimpanan data peristiwa, yang merupakan kumpulan peristiwa yang tidak dapat diubah berdasarkan kriteria yang Anda pilih dengan menerapkan pemilih peristiwa lanj utan. Pemilih yang Anda terapkan ke penyimpanan data peristiwa mengontrol peristiwa mana yang bertahan dan tersedia untuk Anda kueri. Untuk informasi lebih lanjut tentang CloudTrail Danau, lihat Bekerja dengan AWS CloudTrail Danau di Panduan AWS CloudTrail Pengguna.

CloudTrail Penyimpanan dan kueri data acara danau dikenakan biaya. Saat membuat penyimpanan data peristiwa, Anda memilih opsi harga yang ingin Anda gunakan untuk penyimpanan data peristiwa. Opsi penetapan harga menentukan biaya untuk menelan dan menyimpan peristiwa, serta periode retensi default dan maksimum untuk penyimpanan data peristiwa. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

Event data API langsung EBS di CloudTrail

Peristiwa data memberikan informasi tentang operasi sumber daya yang dilakukan pada atau di sumber daya. Ini juga dikenal sebagai operasi bidang data. Peristiwa data sering kali merupakan aktivitas bervolume tinggi. Secara default, CloudTrail tidak mencatat peristiwa data. Ri CloudTrail wayat peristiwa tidak merekam peristiwa data.

Biaya tambahan berlaku untuk peristiwa data. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

Anda dapat mencatat peristiwa data untuk jenis sumber daya API langsung EBS dengan menggunakan CloudTrail konsol, AWS CLI, atau operasi CloudTrail API. Untuk informasi selengkapnya tentang cara mencatat peristiwa data, lihat Pencatatan peristiwa data dengan Konsol Manajemen AWS dan Pencatatan peristiwa data dengan AWS Command Line Interface di Panduan AWS CloudTrail Pengguna.

Anda dapat mencatat operasi API langsung EBS berikut sebagai peristiwa data.

- ListSnapshotBlocks
- ListChangedBlocks
- GetSnapshotBlock
- PutSnapshotBlock

> **Note**
>
> Jika Anda melakukan tindakan pada snapshot yang dibagikan dengan Anda, peristiwa data tidak dikirim ke AWS akun yang memiliki snapshot.

Event manajemen API langsung EBS di CloudTrail

Acara manajemen memberikan informasi tentang operasi manajemen yang dilakukan pada sumber daya di Anda Akun AWS. Ini juga dikenal sebagai operasi bidang kontrol. Secara default, CloudTrail mencatat peristiwa manajemen.

Layanan API langsung EBS mencatat operasi bidang kontrol berikut CloudTrail sebagai peristiwa manajemen.

- StartSnapshot
- CompleteSnapshot

Contoh acara API langsung EBS

Peristiwa mewakili permintaan tunggal dari sumber apa pun dan mencakup informasi tentang operasi API yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi acara tidak muncul dalam urutan tertentu.

Berikut ini adalah contoh CloudTrail peristiwa untuk API langsung EBS.

StartSnapshot


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:root",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
```

```
    "eventTime": "2020-07-03T23:27:26Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "StartSnapshot",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "PostmanRuntime/7.25.0",
    "requestParameters": {
        "volumeSize": 8,
        "clientToken": "token",
        "encrypted": true
    },
    "responseElements": {
        "snapshotId": "snap-123456789012",
        "ownerId": "123456789012",
        "status": "pending",
        "startTime": "Jul 3, 2020 11:27:26 PM",
        "volumeSize": 8,
        "blockSize": 524288,
        "kmsKeyArn": "HIDDEN_DUE_TO_SECURITY_REASONS"
    },
    "requestID": "be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
    "eventID": "6e12345-2a4e-417c-aa78-7594fEXAMPLE",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


CompleteSnapshot


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:root",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
    "eventTime": "2020-07-03T23:28:24Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "CompleteSnapshot",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "192.0.2.0",
```

```
    "userAgent": "PostmanRuntime/7.25.0",
    "requestParameters": {
        "snapshotId": "snap-123456789012",
        "changedBlocksCount": 5
    },
    "responseElements": {
        "status": "completed"
    },
    "requestID": "be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
    "eventID": "6e12345-2a4e-417c-aa78-7594fEXAMPLE",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```

ListSnapshotBlocks

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAT4HPB2AO3JEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/user",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
    "eventTime": "2021-06-03T00:32:46Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "ListSnapshotBlocks",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "111.111.111.111",
    "userAgent": "PostmanRuntime/7.28.0",
    "requestParameters": {
        "snapshotId": "snap-abcdef01234567890",
        "maxResults": 100,
        "startingBlockIndex": 0
    },
    "responseElements": null,
    "requestID": "example6-0e12-4aa9-b923-1555eexample",
    "eventID": "example4-218b-4f69-a9e0-2357dexample",
    "readOnly": true,
    "resources": [
        {
```

```
            "accountId": "123456789012",
            "type": "AWS::EC2::Snapshot",
            "ARN": "arn:aws:ec2:us-west-2::snapshot/snap-abcdef01234567890"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-SHA",
        "clientProvidedHostHeader": "ebs.us-west-2.amazonaws.com"
    }
}
```

ListChangedBlocks

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAT4HPB2AO3JEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/user",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
    "eventTime": "2021-06-02T21:11:46Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "ListChangedBlocks",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "111.111.111.111",
    "userAgent": "PostmanRuntime/7.28.0",
    "requestParameters": {
        "firstSnapshotId": "snap-abcdef01234567890",
        "secondSnapshotId": "snap-9876543210abcdef0",
        "maxResults": 100,
        "startingBlockIndex": 0
    },
    "responseElements": null,
    "requestID": "example0-f4cb-4d64-8d84-72e1bexample",
    "eventID": "example3-fac4-4a78-8ebb-3e9d3example",
```

```
    "readOnly": true,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::EC2::Snapshot",
            "ARN": "arn:aws:ec2:us-west-2::snapshot/snap-abcdef01234567890"
        },
       {
            "accountId": "123456789012",
            "type": "AWS::EC2::Snapshot",
            "ARN": "arn:aws:ec2:us-west-2::snapshot/snap-9876543210abcdef0"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-SHA",
        "clientProvidedHostHeader": "ebs.us-west-2.amazonaws.com"
    }
}
```

GetSnapshotBlock

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAT4HPB2AO3JEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/user",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
    "eventTime": "2021-06-02T20:43:05Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "GetSnapshotBlock",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "111.111.111.111",
    "userAgent": "PostmanRuntime/7.28.0",
    "requestParameters": {
```

```
        "snapshotId": "snap-abcdef01234567890",
        "blockIndex": 1,
        "blockToken": "EXAMPLEiL5E3pMPFpaDWjExM2/mnSKh1mQfcbjwe2mM7EwhrgCdPAEXAMPLE"
    },
    "responseElements": null,
    "requestID": "examplea-6eca-4964-abfd-fd9f0example",
    "eventID": "example6-4048-4365-a275-42e94example",
    "readOnly": true,
    "resources": [
        {
          "accountId": "123456789012",
          "type": "AWS::EC2::Snapshot",
          "ARN": "arn:aws:ec2:us-west-2::snapshot/snap-abcdef01234567890"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-SHA",
        "clientProvidedHostHeader": "ebs.us-west-2.amazonaws.com"
    }
}
```

PutSnapshotBlock

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAT4HPB2AO3JEXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/user",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "user"
    },
    "eventTime": "2021-06-02T21:09:17Z",
    "eventSource": "ebs.amazonaws.com",
    "eventName": "PutSnapshotBlock",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "111.111.111.111",
```

```
    "userAgent": "PostmanRuntime/7.28.0",
    "requestParameters": {
        "snapshotId": "snap-abcdef01234567890",
        "blockIndex": 1,
        "dataLength": 524288,
        "checksum": "exampleodSGvFSb1e3kxWUgbOQ4TbzPurnsfVexample",
        "checksumAlgorithm": "SHA256"
    },
    "responseElements": {
        "checksum": "exampleodSGvFSb1e3kxWUgbOQ4TbzPurnsfVexample",
        "checksumAlgorithm": "SHA256"
    },
    "requestID": "example3-d5e0-4167-8ee8-50845example",
    "eventID": "example8-4d9a-4aad-b71d-bb31fexample",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::EC2::Snapshot",
            "ARN": "arn:aws:ec2:us-west-2::snapshot/snap-abcdef01234567890"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-SHA",
        "clientProvidedHostHeader": "ebs.us-west-2.amazonaws.com"
    }
}
```


Untuk informasi tentang konten CloudTrail rekaman, lihat konten CloudTrail rek aman di Panduan AWS CloudTrail Pengguna.

Pertanyaan yang sering diajukan untuk API langsung EBS

Apakah snapshot dapat diakses menggunakan API langsung EBS jika berstatus tertunda?

Tidak. Snapshot hanya dapat diakses jika memiliki status selesai.

Apakah indeks blok dikembalikan oleh EBS langsung sesuai urutan angka?

Ya. Indeks blok yang dikembalikan bersifat unik, dan dalam urutan numerik.

Dapatkah saya mengirimkan permintaan dengan nilai MaxResults parameter di bawah 100?

Tidak. Nilai MaxResult parameter minimum yang dapat Anda gunakan adalah 100. Jika Anda mengirimkan permintaan dengan nilai MaxResult parameter di bawah 100, dan ada lebih dari 100 blok dalam snapshot, maka API akan mengembalikan setidaknya 100 hasil.

Dapatkah saya menjalankan permintaan API secara bersamaan?

Anda dapat menjalankan permintaan API secara bersamaan. Pastikan untuk mencatat beban kerja lain yang mungkin berjalan di akun untuk menghindari hambatan. Anda juga harus membangun mekanisme coba ulang ke dalam alur kerja API langsung EBS untuk menangani throttling, waktu habis, dan ketidaktersediaan layanan. Untuk informasi selengkapnya, lihat Mengoptimalkan kinerja untuk API langsung EBS.

Tinjau kuota layanan API langsung EBS untuk menentukan permintaan API yang dapat Anda jalankan per detik. Untuk informasi selengkapnya, lihat Titik Akhir dan Kuota Amazon Elastic Block Store dalam Referensi Umum AWS .

Saat menjalankan ListChangedBlocks tindakan, apakah mungkin untuk mendapatkan respons kosong meskipun ada blok di snapshot?

Ya. Jika blok yang diubah berjumlah sedikit di snapshot, respons mungkin kosong tetapi API akan mengembalikan nilai token halaman berikutnya. Gunakan nilai token halaman berikutnya untuk melanjutkan ke halaman hasil berikutnya. Anda dapat mengonfirmasi bahwa Anda telah mencapai halaman terakhir hasil ketika API mengembalikan nilai token halaman berikutnya dari nol.

Jika NextToken parameter ditentukan bersama dengan StartingBlockIndex parameter, mana dari keduanya yang digunakan?

Yang NextToken digunakan, dan StartingBlockIndex yang diabaikan.

Berapa lama token blok dan token berikutnya berlaku?

Token blok berlaku selama tujuh hari, dan token berikutnya berlaku selama 60 menit.

Apakah snapshot terenkripsi didukung?

Ya. Snapshot yang dienkripsi dapat diakses menggunakan API langsung EBS.

Untuk mengakses snapshot terenkripsi, Anda harus memiliki akses ke kunci KMS yang digunakan untuk mengenkripsi snapshot, dan tindakan AWS KMS dekripsi. Lihat Kontrol akses ke API

langsung EBS menggunakan IAM bagian sebelumnya dalam panduan ini untuk AWS KMS kebijakan yang ditetapkan ke pengguna.

Apakah snapshot publik didukung?

Snapshot publik tidak didukung.

Apakah snapshot lokal Amazon EBS didukung? AWS Outposts

Snapshot lokal Amazon EBS aktif tidak AWS Outposts didukung.

Apakah blok snapshot daftar mengembalikan semua indeks blok dan token blok dalam snapshot, atau hanya yang memiliki data yang ditulis ke dalamnya?

Blok tersebut hanya mengembalikan indeks dan token blok yang memiliki data yang ditulis kepadanya.

Dapatkah saya memperoleh riwayat kunjungan API yang dibuat berdasarkan API langsung EBS di akun saya untuk tujuan analisis keamanan dan pemecahan masalah operasional?

Ya. Untuk menerima riwayat panggilan API API langsung EBS yang dilakukan di akun Anda, aktifkan AWS CloudTrail di Konsol Manajemen AWS. Lihat informasi yang lebih lengkap di Log panggilan API langsung EBS menggunakan AWS CloudTrail.

Pulihkan volume EBS yang dihapus, snapshot EBS, dan EBS-backed AMI dengan Recycle Bin

Recycle Bin adalah fitur pemulihan data yang memungkinkan Anda memulihkan volume EBS yang dihapus secara tidak sengaja, snapshot EBS, dan AMI. EBS-backed 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.

Recycle Bin dinilai sebagai kemampuan layanan Amazon Elastic Block Store (Amazon EBS). AWS Layanan apa pun dalam Scope by Compli ance Program (FedRAMP, HIPAA BAA, SOC, dll) yang mencantumkan Amazon EBS juga akan berlaku untuk Recycle Bin.

Topik

- Sumber daya yang didukung
- Bagaimana cara kerja Recycle Bin?
- Pertimbangan untuk Recycle Bin
- Kuota
- Layanan-layanan terkait
- Harga
- Kontrol akses ke Recycle Bin dengan IAM
- Membuat aturan retensi Recycle Bin
- Memperbarui aturan retensi Recycle Bin yang ada
- Mengunci aturan retensi Recycle Bin untuk mencegahnya diperbarui atau dihapus
- Buka kunci aturan retensi Recycle Bin untuk memungkinkannya diperbarui atau dihapus

- Menandai aturan retensi Recycle Bin
- Menghapus aturan retensi Recycle Bin untuk menghentikannya mempertahankan sumber daya
- Pulihkan snapshot yang dihapus dari Recycle Bin
- Pulihkan volume yang dihapus dari Recycle Bin
- Pulihkan AMI yang dihapus dari Recycle Bin
- Pantau Recycle Bin menggunakan Amazon EventBridge
- Memantau Recycle Bin menggunakan AWS CloudTrail
- Titik akhir layanan untuk Recycle Bin
- Buat koneksi pribadi antara VPC dan Recycle Bin

Sumber daya yang didukung

Keranjang Sampah mendukung tipe sumber daya berikut:

- Volume Amazon EBS
- 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.

- Gambar Mesin EBS-backed Amazon Amazon (AMI)

> **Note**
>
> Aturan retensi juga berlaku untuk AMI yang dinonaktifkan.

Bagaimana cara kerja Recycle Bin?

Untuk mengaktifkan dan menggunakan Recycle Bin, Anda harus membuat aturan penyimpanan di AWS Wilayah tempat Anda ingin melindungi sumber daya Anda. Aturan retensi menentukan hal berikut:

- Jenis sumber daya yang ingin Anda lindungi (volume, snapshot, atau AMI).
- Jenis aturan retensi:
  - Tag-level aturan retensi — Aturan retensi ini menggunakan tag sumber daya untuk mengidentifikasi sumber daya yang akan dilindungi. Untuk setiap aturan retensi, Anda menentukan satu atau beberapa pasangan kunci dan nilai tanda. Sumber daya (dari jenis yang ditentukan) yang memiliki setidaknya satu dari kunci tag dan pasangan nilai ini secara otomatis disimpan di Recycle Bin setelah dihapus. Gunakan jenis aturan retensi ini untuk melindungi sumber daya tertentu di akun Anda berdasarkan tagnya.
  - Region-level aturan penyimpanan — Aturan penyimpanan ini, secara default, berlaku untuk semua sumber daya (dari jenis yang ditentukan) di Wilayah, bahkan jika sumber daya tidak diberi tag. Namun, Anda dapat menentukan tag pengecualian untuk mengecualikan sumber daya yang memiliki tag tertentu. Gunakan jenis aturan retensi ini untuk melindungi semua sumber daya dari jenis tertentu di Wilayah.
- Periode retensi untuk mempertahankan sumber daya setelah dihapus. Setelah periode ini berakhir, sumber daya dihapus secara permanen dari Recycle Bin. Periode retensi yang didukung adalah:
  - Volume EBS: 1 - 7 hari
  - Snapshot dan EBS-backed AMI EBS: 1 - 365 hari

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.

Pertimbangan untuk Recycle Bin

Pertimbangan berikut berlaku saat bekerja dengan Keranjang Sampah dan aturan retensi.

Pertimbangan umum

- Sumber daya yang dihapus dipindahkan ke Keranjang Sampah hanya jika sesuai dengan aturan retensi yang ada. Jika Anda menghapus sumber daya yang tidak sesuai dengan aturan retensi, atau jika Anda tidak memiliki aturan penyimpanan pada saat itu, sumber daya tersebut dihapus secara permanen; sumber daya tersebut tidak dipindahkan ke Keranjang Sampah.
-
  > **Important**
  >
  > Aturan retensi mengikuti model konsistensi akhir untuk aturan retensi pertama yang dibuat per jenis sumber daya, per Wilayah di akun Anda. Saat Anda membuat aturan retensi pertama untuk jenis sumber daya di Wilayah, aturan tersebut mungkin tidak aktif dan mulai mempertahankan sumber daya segera. Namun, aturan retensi berikutnya yang Anda buat untuk jenis sumber daya yang sama di Wilayah yang sama akan menjadi aktif dan mulai mempertahankan sumber daya segera.
- 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. Jika perlu, Anda dapat mencegah sumber daya ini masuk ke Recycle Bin setelah dihapus dengan menandai sumber daya tersebut dan kemudian menambahkan tag tersebut sebagai tag pengecualian ke aturan penyimpanan Anda.
- 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 volume

- Volume yang dihapus karena penghentian instans atau penggantian volume root dilindungi oleh Recycle Bin.
- Volume yang gagal dibuat tidak dilindungi oleh Recycle Bin saat dihapus.
- Volume peluncuran instans yang gagal tidak dilindungi oleh Recycle Bin saat dihapus.
- Volume instans terkelola tidak dilindungi oleh Recycle Bin saat dihapus.
- Pembuatan atau modifikasi volume yang sedang berlangsung tidak akan dijeda saat volume memasuki Recycle Bin. Ini berarti Anda masih ditagih sesuai jika volume dibuat dengan Tarif Penyediaan Amazon EBS untuk Inisialisasi Volume.
- Volume di Recycle Bin dihitung dalam kuota Anda dengan cara yang sama seperti volume biasa.
- Volume di Recycle Bin tidak ditagih setelah waktu keluar dari Recycle Bin telah berlalu. Anda tidak dapat memulihkan volume ini tetapi Anda dapat menemukannya jika belum dihapus.
- A `deleteVolume` cara akan dikirim hanya setelah volume dihapus dari Recycle Bin. Peristiwa ini tidak dipancarkan saat volume memasuki Recycle Bin.

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 tersebut dari Recycle Bin, snapshot tersebut tidak lagi dikelola oleh AWS layanan yang membuatnya. Anda harus menghapus snapshot secara manual jika tidak lagi diperlukan.

Pertimbangan untuk AMI

- Hanya EBS-backed AMI Amazon 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 Anda dapat memulihkan AMI dari Recycle Bin, Anda harus terlebih dahulu memulihkan semua snapshot terkait dari Recycle Bin dan memastikan bahwa mereka berada dalam `available` status.
- 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 tersebut dari Recycle Bin, AMI 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 AWS Pencadangan

- Jika AWS Backup menghapus snapshot yang cocok dengan aturan retensi, snapshot tersebut secara otomatis dipertahankan 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.

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 Memantau Recycle Bin menggunakan AWS CloudTrail.

Harga

Tidak ada biaya tambahan untuk menggunakan Keranjang Sampah dan aturan penyimpanan. Untuk informasi selengkapnya, lihat harga Amazon EBS.

- Volume Amazon EBS — Volume di Recycle Bin ditagih dengan tarif yang sama dengan volume reguler di akun Anda.
- Snapshot Amazon EBS — Snapshot di Recycle Bin ditagih dengan tarif yang sama dengan snapshot biasa di akun Anda.
- EBS-backed AMI — AMI di Recycle Bin tidak dikenakan biaya tambahan.

> **Note**
>
> Beberapa sumber daya mungkin masih muncul di konsol Recycle Bin atau di output API AWS CLI dan untuk waktu yang singkat setelah periode retensinya berakhir 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 pelacakan biaya dan tujuan alokasi saat menggunakan AWS Manajemen Penagihan dan Biaya.

- Kunci: `aws:recycle-bin:resource-in-bin`
- Nilai: `true`

Untuk informasi selengkapnya, lihat Tanda alokasi biaya yang Dibuat AWS di Panduan Pengguna AWS Manajemen Penagihan dan Biaya .

Kontrol akses ke Recycle Bin dengan IAM

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.

JSON

```
{
    "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 dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

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 volume di Recycle Bin
- 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

- rbin: kunci Request/ResourceType kondisi
- rbin: kunci Attribute/ResourceType kondisi

`rbin: kunci Request/ResourceType` kondisi

K `rbin:Request/ResourceType` unci 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 `ResourceType` permintaan adalah atau. `EBS_SNAPSHOT` `EC2_IMAGE` Hal ini memungkinkan pengguna utama membuat aturan retensi baru hanya untuk snapshot dan AMI.

JSON

```
{
    "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 `ResourceType` permintaan adalah. `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.

JSON

```
{
    "Version":"2012-10-17",
    "Statement" : [
        {
            "Effect" : "Allow",
            "Action" :[
                "rbin:ListRules"
            ],
            "Resource" : "*",
            "Condition" : {
                "StringEquals" : {
                    "rbin:Request/ResourceType" : "EBS_SNAPSHOT"
                }
            }
        }
    ]
}
```

`rbin: kunci Attribute/ResourceType` kondisi

K `rbin:Attribute/ResourceType` unci kondisi dapat digunakan untuk memfilter akses pada DeleteRule GetRule UpdateRule, LockRule, UnlockRule, TagResource, UntagResource,,, dan ListTagsForResource permintaan berdasarkan nilai `ResourceType` atribut aturan retensi.

Contoh 1 - UpdateRule

Contoh kebijakan IAM berikut mengizinkan prinsipal IAM untuk membuat UpdateRule permintaan hanya jika ResourceType atribut aturan retensi yang diminta adalah atau. EBS_SNAPSHOT EC2_IMAGE Hal ini memungkinkan pengguna utama memperbarui aturan retensi hanya untuk snapshot dan AMI.

JSON

```
{
    "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 aturan retensi yang diminta adalah. EBS_SNAPSHOT Hal ini memungkinkan pengguna utama menghapus aturan retensi hanya untuk snapshot.

JSON

```
{
    "Version":"2012-10-17",
    "Statement" : [
        {
            "Effect" : "Allow",
            "Action" :[
                "rbin:DeleteRule"
```

```
            ],
            "Resource" : "*",
            "Condition" : {
                "StringEquals" : {
                    "rbin:Attribute/ResourceType" : "EBS_SNAPSHOT"
                }
            }
        }
    ]
}
```

Membuat aturan retensi Recycle Bin

Saat membuat aturan retensi, Anda harus menentukan parameter yang diperlukan berikut:

- Jenis sumber daya yang akan dilindungi (volume, snapshot, atau AMI).
- Jenis aturan retensi (tingkat tag atau Region-level). Tag-level aturan hanya melindungi sumber daya yang memiliki tag tertentu. Region-level aturan melindungi semua sumber daya di Wilayah, tetapi dapat mengecualikan sumber daya yang memiliki tag tertentu.
- Periode retensi untuk mempertahankan sumber daya setelah dihapus. Setelah periode ini berakhir, sumber daya dihapus secara permanen dari Recycle Bin. Periode retensi yang didukung adalah:
  - Volume EBS: 1 - 7 hari
  - Snapshot dan EBS-backed AMI EBS: 1 - 365 hari

Anda juga dapat secara opsional menentukan nama aturan dan deskripsi hingga 255 karakter masing-masing, dan tag untuk membantu Anda mengidentifikasi dan mengatur aturan Anda. Kami menyarankan Anda untuk tidak memasukkan informasi identitas pribadi, rahasia, atau sensitif dalam nama, deskripsi, atau tag.

Anda juga dapat mengunci aturan Region-level retensi secara opsional 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.

> **Note**
>
> 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 tingkat tag

1. Buka konsol Recycle Bin di https://console.aws.amazon.com/rbin/home/
2. Di panel navigasi, pilih buat Aturan retensi, lalu pilih Buat aturan retensi.
3. (Opsional) Untuk Nama aturan retensi, masukkan nama deskriptif untuk aturan retensi.
4. (Opsional) Untuk Deskripsi aturan retensi, masukkan deskripsi singkat untuk aturan retensi.
5. Untuk Jenis sumber daya, pilih jenis sumber daya untuk aturan retensi yang akan dilindungi. Aturan retensi hanya akan mempertahankan sumber daya dari tipe ini di Keranjang Sampah.
6. Untuk Pilih sumber daya yang akan dipertahankan, pilih Pertahankan sumber daya yang memiliki tag tertentu.
7. Untuk tag Sumber daya, masukkan kunci tag dan pasangan nilai yang akan digunakan untuk mengidentifikasi sumber daya yang akan disimpan di Recycle Bin. Hanya sumber daya dari jenis yang ditentukan yang memiliki setidaknya satu dari tag yang ditentukan akan dipertahankan oleh aturan retensi.
8. Untuk Periode Retensi, masukkan jumlah hari untuk menyimpan sumber daya yang dihapus di Recycle Bin.
9. Pilih Buat aturan retensi.

Untuk membuat aturan Region-level retensi

1. Buka konsol Recycle Bin di https://console.aws.amazon.com/rbin/home/
2. Di panel navigasi, pilih buat Aturan retensi, lalu pilih Buat aturan retensi.
3. (Opsional) Untuk Nama aturan retensi, masukkan nama deskriptif untuk aturan retensi.

4. (Opsional) Untuk Deskripsi aturan retensi, masukkan deskripsi singkat untuk aturan retensi.
5. Untuk Jenis sumber daya, pilih jenis sumber daya untuk aturan retensi yang akan dilindungi. Aturan retensi hanya akan mempertahankan sumber daya dari tipe ini di Keranjang Sampah.
6. Untuk Pilih sumber daya yang akan dipertahankan, pilih Per tahankan semua sumber daya.
7. (Opsional) Untuk mengecualikan sumber daya yang memiliki tag tertentu, untuk tag Peng ecualian, masukkan hingga lima kunci tag dan pasangan nilai yang akan digunakan untuk mengidentifikasi sumber daya yang akan dikecualikan. Sumber daya yang memiliki salah satu tag ini diabaikan oleh aturan retensi.
8. Untuk Periode Retensi, masukkan jumlah hari untuk menyimpan sumber daya yang dihapus di Recycle Bin.
9. (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 Recycle Bin untuk mencegahnya diperbarui atau dihapus

   Agar aturan retensi tidak terkunci, untuk Pengaturan penguncian aturan, tetap pilih Buka kunci. Aturan retensi yang tidak terkunci dapat diubah atau dihapus kapan saja.

   > **Note**
   >
   > Anda tidak dapat mengunci aturan Region-level retensi yang memiliki tag pengecualian.

10. 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`, tent `EBS_VOLUME` ukan untuk volume, `EBS_SNAPSHOT` untuk snapshot, atau `EC2_IMAGE` untuk AMI. Untuk membuat aturan retensi tingkat tag, for`--resource-tags`, tentukan tag yang akan digunakan untuk mengidentifikasi sumber daya yang akan dipertahankan. Untuk membuat aturan Region-level retensi, hilangkan, dan tentukan `--resource-tags` secara opsional`--`

`exclude-resource-tags`, untuk mengecualikan sumber daya yang memiliki tag tertentu. Untuk mengunci aturan Region-level retensi, sertakan`--lock-configuration`, dan tentukan periode penundaan buka kunci dalam hari.

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=number_of_days,RetentionPeriodUnit=DAYS \
--resource-type EBS_VOLUME|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 \
--exclude-resource-tags ResourceTagKey=tag_key,ResourceTagValue=tag_value
```

Contoh 1

Perintah contoh berikut membuat aturan Region-level retensi tidak ter kunci yang menyimpan semua snapshot yang dihapus untuk jangka waktu beberapa 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

Perintah contoh berikut membuat aturan Region-level retensi terkunci yang menyimpan semua snapshot yang dihapus selama beberapa 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}'
```

Contoh 4

Perintah contoh berikut membuat aturan Region-level retensi tidak ter kunci yang menyimpan semua snapshot yang dihapus, kecuali snapshot yang diberi tag`purpose:testing`, untuk jangka waktu beberapa hari. 7

```
aws rbin create-rule \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Match only production snapshots" \
--exclude-resource-tags ResourceTagKey=purpose,ResourceTagValue=testing
```

Memperbarui aturan retensi Recycle Bin yang ada

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 Recycle Bin di 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_VOLUME` untuk volume, `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_VOLUME|EBS_SNAPSHOT|EC2_IMAGE \
--description "rule_description"
```

Contoh

Perintah contoh berikut memperbarui aturan retensi `6lsJ2Fa9nh9` untuk mempertahankan semua snapshot selama 7 hari dan memperbarui deskripsinya.

```
aws rbin update-rule \
--identifier 6lsJ2Fa9nh9 \
--retention-period RetentionPeriodValue=7,RetentionPeriodUnit=DAYS \
--resource-type EBS_SNAPSHOT \
--description "Retain for three weeks"
```

Mengunci aturan retensi Recycle Bin untuk mencegahnya diperbarui atau dihapus

Recycle Bin memungkinkan Anda Region-level mengunci aturan retensi kapan saja.

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 berkahir.

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 tidak dapat mengunci aturan retensi tingkat tag, atau aturan Region-level retensi yang memiliki tag pengecualian.
- 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 Region-level retensi menggunakan salah satu metode berikut.

Recycle Bin console

Untuk mengunci aturan retensi

1. Buka konsol Recycle Bin 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 `6lsJ2Fa9nh9` dan menetapkan periode penundaan pembukaan kunci menjadi 15 hari.

```
aws rbin lock-rule \
--identifier 6lsJ2Fa9nh9 \
--lock-configuration 'UnlockDelay={UnlockDelayUnit=DAYS,UnlockDelayValue=15}'
```

Buka kunci aturan retensi Recycle Bin untuk memungkinkannya diperbarui atau dihapus

Anda tidak dapat memodifikasi atau menghapus aturan retensi yang terkunci. Jika perlu memodifikasi aturan retensi yang terkunci, Anda harus membukanya terlebih dahulu. Setelah membuka kunci aturan penyimpanan, Anda harus menunggu periode penundaan buka kunci berakhir sebelum Anda dapat mengubah 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 Region-level retensi terkunci menggunakan salah satu metode berikut.

Recycle Bin console

Untuk membuka kunci aturan retensi

1. Buka konsol Recycle Bin 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 `6lsJ2Fa9nh9`

```
aws rbin unlock-rule \
--identifier 6lsJ2Fa9nh9
```

Menandai aturan retensi Recycle Bin

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 Recycle Bin di 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 Recycle Bin 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 Recycle Bin 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
```

Menghapus aturan retensi Recycle Bin untuk menghentikannya mempertahankan sumber daya

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 Recycle Bin di 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 `6lsJ2Fa9nh9`.

```
aws rbin delete-rule --identifier 6lsJ2Fa9nh9
```

Pulihkan snapshot yang dihapus dari Recycle Bin

Topik ini menjelaskan cara memulihkan snapshot Amazon EBS dari Recycle Bin.

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.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam Menambahkan izin ke pengguna (konsol) 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 Recycle Bin 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 Recycle Bin 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 restore- AWS CLI 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
}
```

Pulihkan volume yang dihapus dari Recycle Bin

Topik ini menjelaskan cara memulihkan volume Amazon EBS dari Recycle Bin.

Topik

- Izin untuk bekerja dengan volume di Recycle Bin
- Melihat volume di Recycle Bin
- Kembalikan volume dari Recycle Bin

Izin untuk bekerja dengan volume di Recycle Bin

Secara default, pengguna tidak memiliki izin untuk bekerja dengan volume yang ada di Recycle Bin. 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 volume yang ada di Recycle Bin, pengguna harus memiliki izin berikut:

- `ec2:ListVolumesInRecycleBin`
- `ec2:RestoreVolumeFromRecycleBin`

Untuk mengelola tag untuk volume di Recycle Bin, 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": [
    {
      "Sid": "AllowRecycleBinVolumeOperations",
      "Effect": "Allow",
      "Action": [
        "ec2:ListVolumesInRecycleBin",
        "ec2:RestoreVolumeFromRecycleBin"
      ],
      "Resource": "arn:aws:ec2:*:123456789012:volume/*"
    },
    {
      "Sid": "AllowVolumeTagOperations",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2:DeleteTags",
        "ec2:DescribeTags"
      ],
```

```
      "Resource": "arn:aws:ec2:*:123456789012:volume/*"
    }
  ]
}
```

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam Menambahkan izin ke pengguna (konsol) 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 volume di Recycle Bin

Saat volume berada di Recycle Bin, Anda dapat melihat informasi terbatas tentangnya, termasuk:

- ID volume.
- Ukuran volume.
- Jenis volume.
- Tanggal dan waktu ketika volume dihapus dan masuk ke Recycle Bin.
- Tanggal dan waktu ketika periode retensi kedaluwarsa. Volume akan dihapus secara permanen dari Recycle Bin saat ini.

Anda dapat melihat volume di Recycle Bin menggunakan salah satu metode berikut.

Recycle Bin console

Untuk melihat volume di Recycle Bin menggunakan konsol

1. Buka konsol Recycle Bin di https://console.aws.amazon.com/rbin/home/
2. Di panel navigasi, pilih Keranjang Sampah.
3. Grid mencantumkan semua volume yang saat ini ada di Recycle Bin. Untuk melihat detail untuk volume tertentu, pilih di grid dan pilih Tind akan, Lihat detail.

AWS CLI

Untuk melihat volume di Recycle Bin menggunakan AWS CLI

Gunakan perintah list-volumes-in-recycle-bin AWS CLI . Sertakan `--volume-id` opsi untuk melihat volume tertentu. Atau hilangkan `--volume-id` opsi untuk melihat semua volume di Recycle Bin.

```
aws ec2 list-volumes-in-recycle-bin --volume-id volume_id
```

Misalnya, perintah berikut memberikan informasi tentang volume `vol-01234567890abcdef` di Recycle Bin.

```
aws ec2 list-volumes-in-recycle-bin --volume-id vol-01234567890abcdef
```

Contoh output:

```
{
    "VolumeRecycleBinInfo": [
        {
            "VolumeId": "vol-01234567890abcdef",
            "RecycleBinEnterTime": "2021-12-01T13:00:00.000Z",
            "RecycleBinExitTime": "2021-12-08T13:00:00.000Z"
        }
    ]
}
```

Kembalikan volume dari Recycle Bin

Anda tidak dapat menggunakan volume dengan cara apa pun saat berada di Recycle Bin. Untuk menggunakan volume, Anda harus mengembalikannya terlebih dahulu. Saat Anda mengembalikan volume dari Recycle Bin, volume segera tersedia untuk digunakan, dan dihapus dari Recycle Bin. Anda dapat menggunakan volume yang dipulihkan dengan cara yang sama seperti Anda menggunakan volume lain di akun Anda.

Anda dapat mengembalikan volume dari Recycle Bin menggunakan salah satu metode berikut.

Recycle Bin console

Untuk mengembalikan volume dari Recycle Bin menggunakan konsol

1. Buka konsol Recycle Bin di https://console.aws.amazon.com/rbin/home/
2. Di panel navigasi, pilih Keranjang Sampah.
3. Grid mencantumkan semua volume yang saat ini ada di Recycle Bin. Pilih volume yang akan dipulihkan dan pilih Pu lihkan.
4. Saat diminta, pilih Pulihkan.

AWS CLI

Untuk mengembalikan volume yang dihapus dari Recycle Bin menggunakan AWS CLI

Gunakan perintah restore- AWS CLI volume-from-recycle-bin. Untuk`--volume-id`, tentukan ID volume yang akan dipulihkan.

```
aws ec2 restore-volume-from-recycle-bin --volume-id volume_id
```

Misalnya, perintah berikut mengembalikan volume `vol-01234567890abcdef` dari Recycle Bin.

```
aws ec2 restore-volume-from-recycle-bin --volume-id vol-01234567890abcdef
```

Contoh output:

```
{
    "VolumeId": "vol-01234567890abcdef",
```

```
    "State": "available",
    "Size": 100,
    "VolumeType": "gp3",
    "AvailabilityZone": "us-east-1a",
    "CreateTime": "2021-12-01T13:00:00.000000+00:00",
    "Encrypted": false
}
```

Pulihkan AMI yang dihapus dari Recycle Bin

Topik ini menjelaskan cara memulihkan EBS-backed AMI Amazon dari Recycle Bin.

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.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam Menambahkan izin ke pengguna (konsol) 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 Recycle Bin di console.aws.amazon. com/rbin/rumah /.
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 perintah list-images-in-recycle-bin AWS CLI . 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 Recycle Bin di console.aws.amazon. com/rbin/rumah /.
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 perintah restore- AWS CLI 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 Recycle Bin menggunakan Amazon EventBridge

Recycle Bin mengirimkan acara ke Amazon EventBridge untuk tindakan yang dilakukan pada aturan retensi. Dengan EventBridge, Anda dapat menetapkan aturan yang memulai tindakan terprogram sebagai respons terhadap peristiwa ini. Misalnya, Anda dapat membuat EventBridge aturan yang mengirimkan pemberitahuan ke email Anda saat aturan retensi dibuka kuncinya dan memasuki periode penundaan buka kuncinya. 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 peristiwa Amazon di Panduan EventBridge Pengguna Amazon.

Untuk informasi lebih lanjut tentang Amazon EventBridge, lihat Apa itu Amazon EventBridge? dalam Panduan EventBridge Pengguna Amazon.

Peristiwa

- RuleLocked
- RuleChangeAttempted

- RuleUnlockScheduled
- RuleUnlockingNotice
- 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"
 }
 }
```

RuleChangeAttempted

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"
 }
 }
```

RuleUnlockScheduled

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",
 }
```

```
 }
```

RuleUnlockingNotice

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"
}
}
```

Memantau 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 di Ri wayat acara. Anda dapat menggunakan informasi yang dikumpulkan oleh CloudTrail untuk menentukan permintaan yang dibuat ke Recycle Bin, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan itu dibuat, dan detail tambahan.

Untuk informasi selengkap CloudTrail nya, lihat Panduan AWS CloudTrail Pengguna.

Informasi Recycle Bin di CloudTrail

CloudTrail diaktifkan pada AWS akun Anda saat Anda membuat akun. Ketika aktivitas acara yang didukung terjadi di Recycle Bin, aktivitas tersebut direkam dalam suatu CloudTrail acara bersama dengan peristiwa AWS layanan lainnya di ri wayat acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat Melihat Acara dengan Ri CloudTrail wayat Acara.

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk Recycle Bin, buat jejak. Jej ak memungkinkan CloudTrail untuk mengirimkan file log ke bucket S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wil AWS ayah.

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 bertindak berdasarkan 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 manajemen logging, lihat Peristiwa pengelolaan logging untuk jalur 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 gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi lebih lanjut, 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 apa 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.

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 botocore/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 botocore/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 botocore/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 botocore/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"
  }
}
},
"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 botocore/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"
}
}
```


Titik akhir layanan untuk Recycle Bin

Endpoint adalah URL yang berfungsi sebagai titik masuk untuk layanan AWS web. Recycle Bin mendukung jenis titik akhir berikut:

- Titik akhir IPv4
- Dual-stack Endpoint yang mendukung IPv4 dan IPv6
- Titik akhir FIPS

Saat Anda membuat permintaan, Anda dapat menentukan titik akhir dan Wilayah yang akan digunakan. Jika Anda tidak menentukan titik akhir, titik akhir IPv4 digunakan secara default. Untuk menggunakan tipe titik akhir yang berbeda, Anda harus menentukannya dalam permintaan Anda. Untuk contoh cara melakukannya, lihat Menentukan titik akhir.

Untuk Recycle Bin, lihat titik akhir Recycle Bin di Referensi Umum Amazon Web Services.

Topik

- Titik akhir IPv4
- Dual-stack Titik akhir (IPv4 dan IPv6)
- Titik akhir FIPS
- Menentukan titik akhir

Titik akhir IPv4

Titik akhir IPv4 hanya mendukung lalu lintas IPv4. Titik akhir IPv4 tersedia untuk semua Wilayah.

Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Nama titik akhir menggunakan konvensi penamaan berikut:

- rbin. *region*.amazonaws.com

Misalnya, titik akhir IPv4 untuk Wilayah AS Timur (Virginia Utara) adalah`rbin.us-east-1.amazonaws.com`.

Dual-stack Titik akhir (IPv4 dan IPv6)

Dual-stack Endpoint mendukung lalu lintas IPv4 dan IPv6. Dual-stack titik akhir tersedia untuk semua Wilayah.

Untuk menggunakan IPv6, Anda harus menggunakan titik akhir tumpukan ganda. Saat Anda membuat permintaan ke titik akhir tumpukan ganda, URL titik akhir memutuskan ke alamat IPv6 atau IPv4, tergantung pada protokol yang digunakan oleh jaringan dan klien Anda.

Anda harus menentukan Wilayah sebagai bagian dari nama titik akhir. Dual-stack nama titik akhir menggunakan konvensi penamaan berikut:

- `rbin.`*region*`.api.aws`

Misalnya, titik akhir tumpukan ganda untuk Wilayah AS Timur (Virginia Utara) adalah`rbin.us-east-1.api.aws`.

Titik akhir FIPS

Recycle Bin menyediakan titik FIPS-validated akhir IPv4 dan dual-stack (IPv4 dan IPv6) untuk Wilayah berikut:

- `us-east-1` — AS Timur (Virginia Utara)
- `us-east-2` — AS Timur (Ohio)
- `us-west-1` — AS Barat (California Utara)
- `us-west-2` — AS Barat (Oregon)

- `ca-central-1` – Kanada (Pusat)
- `ca-west-1`— Kanada Barat (Calgary)
- `us-gov-east-1` — AWS GovCloud (US-East)
- `us-gov-west-1` — AWS GovCloud (US-West)

Titik akhir IPv4 FIPS menggunakan konvensi penamaan berikut: `rbin-fips.region.amazonaws.com`. Misalnya, titik akhir IPv4 FIPS untuk Wilayah Timur AS (Virginia Utara) adalah. `rbin-fips.us-east-1.amazonaws.com`

Titik akhir tumpukan ganda FOPS menggunakan konvensi penamaan berikut: `rbin-fips.region.api.aws`. Misalnya, titik akhir tumpukan ganda FIPS untuk Wilayah Timur AS (Virginia Utara) adalah. `rbin-fips.us-east-1.api.aws`

Menentukan titik akhir

Contoh berikut menunjukkan cara menentukan titik akhir untuk Wilayah `us-east-2` menggunakan AWS CLI.

- Dual-stack

```
aws rbin get-rule \
--identifier rule_id \
--endpoint-url https://rbin.us-east-2.api.aws
```

- IPv4

```
aws rbin get-rule \
--identifier rule_id \
--endpoint-url https://rbin.us-east-2.amazonaws.com
```

Buat koneksi pribadi antara VPC dan Recycle Bin

Anda dapat membuat koneksi pribadi antara VPC dan Recycle Bin dengan membuat antarmuka titik akhir VPC, didukung oleh. AWS PrivateLink Anda dapat mengakses Recycle Bin seolah-olah berada di VPC Anda, tanpa menggunakan gateway internet, perangkat NAT, koneksi VPN, atau Direct Connect koneksi. Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan Recycle Bin.

Kami membuat antarmuka jaringan endpoint di setiap subnet yang Anda aktifkan untuk titik akhir antarmuka.

Untuk informasi selengkapnya, lihat AWS Layanan Ak AWS PrivateLink ses melalui AWS PrivateLink Panduan.

Buat titik akhir VPC antarmuka untuk Recycle Bin

Anda dapat membuat titik akhir VPC untuk Recycle Bin menggunakan konsol Amazon VPC atau. AWS CLI Untuk informasi selengkapnya, lihat Membuat titik akhir VPC di Panduan Pengguna AWS PrivateLink .

Buat titik akhir VPC untuk Recycle Bin menggunakan nama layanan berikut: `com.amazonaws.region.rbin`

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API ke Recycle Bin menggunakan nama DNS default untuk Wilayah, misalnya,`rbin.us-east-1.amazonaws.com`.

Membuat kebijakan titik akhir VPC untuk Recycle Bin

Secara default, akses penuh ke Recycle Bin diizinkan melalui titik akhir. Anda dapat mengontrol akses ke titik akhir antarmuka menggunakan kebijakan titik akhir VPC. Anda dapat melampirkan kebijakan titik akhir ke titik akhir VPC yang mengontrol akses ke Recycle Bin. Kebijakan titik akhir menentukan informasi berikut:

- Kepala sekolah yang dapat melakukan tindakan.
- T ind akan yang bisa dilakukan.
- Sumber daya di mana tindakan dapat dilakukan.

Untuk informasi selengkapnya, lihat Mengendalikan Akses ke Layanan dengan Titik Akhir VPC dalam Panduan Pengguna VPC Amazon.

```
{
    "Statement": [
    {
        "Effect": "Allow",
        "Action": "rbin:*",
        "Resource": "*",
```

```
        "Principal": "*"
    },
    {
        "Effect": "Deny",
        "Action": "rbin:DeleteRule",
        "Resource": "*",
        "Principal": "*",
        "Condition": {
            "StringEquals" : {
                "rbin:Attribute/ResourceType": "EBS_SNAPSHOT"
            }
        }
    }]
}
```

Keamanan di Amazon EBS

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 dari cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Third-party auditor secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program Kepatuhan Program AWS Kepatuhan Program AWS . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon Elastic Block Store, lihat AWS Layanan dalam Lingkup berdasarkan AWS Layanan Program Kepatuhan .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Amazon EBS. Topik berikut menunjukkan cara mengonfigurasi Amazon EBS untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Amazon EBS Anda.

Topik

- Perlindungan data di Amazon EBS
- Manajemen identitas dan akses untuk Amazon EBS
- Validasi kepatuhan untuk Amazon EBS
- Ketahanan data di Amazon EBS

Perlindungan data di Amazon EBS

Model tanggung jawab AWS https://aws.amazon.com/compliance/shared-responsibility-model/ berlaku untuk perlindungan data di Amazon Elastic Block Store. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua

AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat FAQ Privasi Data Ket . Untuk informasi tentang perlindungan data di Eropa, lihat Pusat Peraturan Umum Perlindungan Data (GDPR).

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredenSIAL dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang menggunakan CloudTrail jalur untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di Panduan AWS CloudTrail Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan Amazon EBS 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.

Topik

- Keamanan data Amazon EBS
- Enkripsi saat istirahat dan dalam transit
- Manajemen kunci KMS

Keamanan data Amazon EBS

Volume Amazon EBS disajikan kepada Anda sebagai perangkat blok mentah yang tidak terformat. Perangkat ini adalah perangkat logis yang dibuat pada infrastruktur EBS. Layanan Amazon EBS memastikan bahwa perangkat secara logis kosong (yaitu, blok mentah dinol atau berisi data pseudorandom kriptografis) sebelum pelanggan menggunakan atau digunakan kembali.

Jika prosedur Anda memerlukan penghapusan semua data menggunakan metode tertentu, seperti yang dirinci dalam DoD 5220.22-M (Manual Operasi Program Keamanan Industri Nasional) atau NIST 800-88 (Pedoman untuk Sanitasi Media), Anda dapat melakukannya di Amazon EBS. Anda dapat menghapus data sebelum atau sesudah digunakan, atau keduanya tergantung pada persyaratan prosedur. Aktivitas tingkat blok tersebut akan tercermin ke media penyimpanan yang mendasarinya dalam layanan Amazon EBS tersebut.

Enkripsi saat istirahat dan dalam transit

Enkripsi Amazon EBS adalah solusi enkripsi yang memungkinkan Anda mengenkripsi volume Amazon EBS dan snapshot Amazon EBS menggunakan kunci kriptografi. AWS Key Management Service Operasi enkripsi EBS terjadi pada server yang meng-host instans Amazon EC2. Ini memastikan keamanan data-at-rest dan data-in-transit antara instance dan volume terlampirnya dan snapshot berikutnya. Untuk informasi selengkapnya, lihat Enkripsi EBS Amazon.

Manajemen kunci KMS

Saat Anda membuat volume atau snapshot Amazon EBS terenkripsi, Anda menentukan AWS Key Management Service kunci. Secara default, Amazon EBS menggunakan kunci KMS AWS terkelola untuk Amazon EBS di akun dan Wilayah Anda (). `aws/ebs` Namun, Anda dapat menentukan kunci KMS yang dikelola pelanggan yang Anda buat dan kelola. Menggunakan kunci KMS yang dikelola pelanggan memberi Anda lebih banyak fleksibilitas, termasuk kemampuan untuk membuat, memutar, dan menonaktifkan kunci KMS.

Untuk menggunakan kunci KMS yang dikelola pelanggan, Anda harus memberikan izin kepada pengguna untuk menggunakan kunci KMS. Untuk informasi selengkapnya, lihat Izin untuk pengguna.

> **Important**
>
> Amazon EBS hanya mendukung kunci KMS simetris. Anda tidak dapat menggunakan kunci KMS asimetris untuk mengenkripsi volume dan snapshot Amazon EBS. Untuk bantuan menentukan apakah kunci KMS simetris atau asimetris, lihat Mengidentifikasi kunci KMS asimetris.

Untuk setiap volume, Amazon EBS meminta AWS KMS untuk membuat kunci data unik yang dienkripsi di bawah kunci KMS yang Anda tentukan. Amazon EBS menyimpan kunci data terenkripsi dengan volume. Kemudian, saat Anda melampirkan volume ke instans Amazon EC2, Amazon EBS memanggil AWS KMS untuk mendekripsi kunci data. Amazon EBS menggunakan kunci data plaintext dalam memori hypervisor untuk mengenkripsi semua I/O ke volume. Lihat informasi yang lebih lengkap di Cara kerja enkripsi Amazon EBS.

Manajemen identitas dan akses untuk Amazon EBS

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan di otorisasi (memiliki izin) untuk menggunakan sumber daya Amazon EBS. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Cara kerja Amazon EBS dengan IAM
- Contoh kebijakan IAM untuk Amazon EBS
- Memecahkan masalah otorisasi Amazon EBS

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda berdasarkan peran Anda:

- Pengguna layanan - minta izin dari administrator Anda jika Anda tidak dapat mengakses fitur (lihat Memecahkan masalah otorisasi Amazon EBS)
- Administrator layanan - tentukan akses pengguna dan mengirimkan permintaan izin (lihat Cara kerja Amazon EBS dengan IAM)
- Administrator IAM - tulis kebijakan untuk mengelola akses (lihat Contoh kebijakan IAM untuk Amazon EBS)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredentif identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengambil peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredenSIAL dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), otentikasi masuk tunggal, atau kredenSIAL. Google/ Facebook Untuk informasi selengkapnya tentang cara masuk, lihat Cara masuk ke Akun AWS Anda dalam Panduan Pengguna AWS Sign-In .

Untuk akses terprogram, AWS menyediakan SDK dan CLI untuk menandatangani permintaan kriptografis. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensial pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas terfederasi

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredenSIAL sementara.

Identi tas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredenSIAL dari sumber identitas. Identitas terfederasi mengambil peran yang memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami menyarankan AWS IAM Identity Center. Untuk informasi selengkapnya, lihat Apa itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensial sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Memerlukan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses AWS menggunakan kredenSIAL sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan mempermudah pengelolaan izin untuk pengguna dalam jumlah besar. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensial sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna terfederasi, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya. AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan principal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diambil oleh pengguna. Kebijakan IAM mendefinisikan izin terlepas dari metode yang Anda gunakan untuk melakukan operasinya.

Identity-based kebijakan

Identity-based kebijakan adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang bisa dilakukan oleh identitas tersebut, terhadap sumber daya yang mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan yang dikelola pelanggan dalam Panduan Pengguna IAM.

Identity-based kebijakan dapat berupa kebijakan sebaris (disematkan langsung ke dalam satu identitas) atau kebijakan ter kelola (kebijakan mandiri yang dilampirkan ke beberapa identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan kebijakan inline, lihat Pilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Resource-based kebijakan

Resource-based kebijakan adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

Resource-based kebijakan adalah kebijakan sebaris yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat mengatur izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batasan izin – Menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCP) – Menentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam Panduan Pengguna AWS Organizations .

- Kebijakan kontrol sumber daya (RCP) – Menetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCP) dalam Panduan Pengguna AWS Organizations .
- Kebijakan sesi – Kebijakan lanjutan yang diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna terfederasi. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan ketika beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Cara kerja Amazon EBS dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Amazon EBS, pelajari fitur IAM apa yang tersedia untuk digunakan dengan Amazon EBS.

Fitur IAM yang dapat Anda gunakan dengan Amazon Elastic Block Store

Fitur IAM	Dukungan Amazon EBS
Identity-based kebijakan	Ya
Resource-based kebijakan	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci kondisi kebijakan	Ya
ACL	Tidak
ABAC (tanda dalam kebijakan)	Parsial
Kredensial sementara	Ya
Izin principal	Ya

Fitur IAM	Dukungan Amazon EBS
Peran layanan	Ya
Service-linked peran	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang cara kerja Amazon EBS dan AWS layanan lainnya dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Identity-based kebijakan untuk Amazon EBS

Mendukung kebijakan berbasis identitas: Ya

Identity-based kebijakan adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke identitas, seperti pengguna IAM, grup pengguna, atau peran. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Identity-based contoh kebijakan untuk Amazon EBS

Untuk melihat contoh kebijakan berbasis identitas Amazon EBS, lihat. Contoh kebijakan IAM untuk Amazon EBS

Resource-based kebijakan dalam Amazon EBS

Mendukung kebijakan berbasis sumber daya: Tidak

Resource-based kebijakan adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan

oleh principal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan principal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai principal dalam kebijakan berbasis sumber daya. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk Amazon EBS

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan Amazon EBS, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon EC2 dan Tindakan, sumber daya, dan kunci kondisi untuk Amazon EBS di Referensi Otorisasi Layanan.

Tindakan kebijakan di Amazon EBS menggunakan awalan `ec2` atau `ebs` awalan sebelum tindakan.

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
  "ec2:action1",
  "ec2:action2"
]
```

Untuk melihat contoh kebijakan berbasis identitas Amazon EBS, lihat. Contoh kebijakan IAM untuk Amazon EBS

Sumber daya kebijakan untuk Amazon EBS

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Beberapa tindakan Amazon EBS API mendukung beberapa sumber daya. Untuk menentukan beberapa sumber daya dalam satu pernyataan, pisahkan ARN dengan koma. Misalnya, `DescribeVolumes` mengakses vol-01234567890abcdef dan vol-09876543210fedcba, sehingga kepala sekolah harus memiliki izin untuk mengakses kedua sumber daya.

```
"Resource": [
  "arn:aws:ec2:us-east-1:123456789012:volume/vol-01234567890abcdef",
  "arn:aws:ec2:us-east-1:123456789012:volume/vol-09876543210fedcba"
]
```

Kunci kondisi kebijakan untuk Amazon EBS

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Condition` menentukan ketika pernyataan dieksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Misalnya, kondisi berikut memungkinkan prinsipal untuk melakukan tindakan pada volume hanya jika jenis volume adalahgp2.

```
"Condition":{
  "StringLikeIfExists":{
```

```
    "ec2:VolumeType":"gp2"
  }
}
```

Untuk melihat daftar kunci kondisi Amazon EBS, lihat Tindakan, sumber daya, dan kunci kondisi di Refer ensi Otorisasi Layanan.

ACL di Amazon EBS

Mendukung ACL: Tidak

Daftar kontrol akses (ACL) mengendalikan principal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL serupa dengan kebijakan berbasis sumber daya, meskipun kebijakan tersebut tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan Amazon EBS

Mendukung ABAC (tag dalam kebijakan): Sebagian

Attribute-based kontrol akses (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan atribut yang disebut tag. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi ketika tag kepala sekolah cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredenSIAL sementara dengan Amazon EBS

Mendukung kredensial sementara: Ya

KredenSIAL sementara menyediakan akses jangka pendek ke AWS sumber daya dan dibuat secara otomatis saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda

secara dinamis membuat kredenSIAL sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM dalam Panduan Pengguna IAM.

Cross-service izin utama untuk Amazon EBS

Mendukung sesi akses terusan (FAS): Ya

Sesi akses maju (FAS) menggunakan izin dari kepala sekolah yang memanggil Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses terusan.

Peran layanan untuk Amazon EBS

Mendukung peran layanan: Ya

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak fungsionalitas Amazon EBS. Edit peran layanan hanya jika Amazon EBS memberikan panduan untuk melakukannya.

Service-linked peran untuk Amazon EBS

Mendukung peran terkait layanan: Tidak

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan dapat mengambil peran untuk melakukan tindakan atas nama Anda. Service-linked peran muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Temukan layanan dalam tabel yang menyer Yes takan kolom Service-linked peran. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Contoh kebijakan IAM untuk Amazon EBS

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya Amazon EBS. Mereka juga tidak dapat melakukan tugas dengan menggunakan Konsol Manajemen AWS, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian akan dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Topik

- Praktik terbaik kebijakan
- Izinkan pengguna menggunakan konsol Amazon EBS
- Mengizinkan pengguna melihat izin mereka sendiri
- Izinkan pengguna bekerja dengan volume
- Izinkan pengguna bekerja dengan snapshot

Praktik terbaik kebijakan

Identity-based kebijakan menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya Amazon EBS di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Memulai kebijakan AWS terkelola dan beralih ke izin hak istimewa terendah — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan terkel AWS ola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi

tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.

- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan kondisi untuk memberikan akses ke tindakan layanan jika digunakan melalui tindakan tertentu Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) — Jika Anda memiliki skenario yang memerlukan pengguna IAM atau pengguna root Akun AWS, aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Izinkan pengguna menggunakan konsol Amazon EBS

Untuk mengakses konsol Amazon Elastic Block Store, Anda harus memiliki seperangkat izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya Amazon EBS di. Akun AWS Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan konsol Amazon EBS, lampirkan juga Amazon EBS *ConsoleAccess* atau kebijakan terkel *ReadOnly* AWS ola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau secara terprogram menggunakan API AWS CLI or AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Izinkan pengguna bekerja dengan volume

Contoh

- Contoh: Melampirkan dan melepaskan lampiran volume
- Contoh: Membuat volume
- Contoh: Membuat volume dengan tanda
- Contoh: Bekerja dengan volume menggunakan konsol Amazon EC2

Contoh: Melampirkan dan melepaskan lampiran volume

Ketika tindakan API memerlukan pemerintah untuk menentukan beberapa sumber daya, Anda harus membuat pernyataan kebijakan yang memungkinkan para pengguna mengakses semua sumber daya yang diperlukan. Jika Anda harus menggunakan elemen `Condition` dengan satu atau beberapa sumber daya ini, maka Anda harus membuat beberapa pernyataan seperti yang ditunjukkan dalam contoh berikut ini.

Kebijakan berikut memungkinkan para pengguna untuk melampirkan volume dengan tanda "`volume_user`=iam-user-name" ke instans dengan tanda "`department=dev`", dan untuk melepaskan lampiran volume tersebut dari instans tersebut. Jika Anda melampirkan kebijakan ini ke grup IAM, variabel kebijakan `aws:username` akan memberikan izin kepada setiap pengguna yang ada dalam grup untuk melampirkan atau melepaskan volume dari instans dengan nama tanda `volume_user` yang menjadikan nama pengguna IAM-nya sebagai nilai.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachVolume",
                "ec2:DetachVolume"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
            "Condition": {
```

```
                "StringEquals": {
                    "aws:ResourceTag/department": "dev"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachVolume",
                "ec2:DetachVolume"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/volume_user": "${aws:username}"
                }
            }
        }
    ]
}
```

Contoh: Membuat volume

Kebijakan berikut memungkinkan pengguna untuk menggunakan tindakan API CreateVolume. Pengguna diperbolehkan untuk membuat volume hanya jika volume tersebut dienkripsi dan hanya jika ukuran volume kurang dari 20 GiB.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateVolume"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "NumericLessThan": {
                    "ec2:VolumeSize": "20"
```

```
                },
                "Bool": {
                    "ec2:Encrypted": "true"
                }
            }
        }
    ]
}
```

Contoh: Membuat volume dengan tanda

Kebijakan berikut mencakup kunci syarat `aws:RequestTag` yang mengharuskan para pengguna untuk menandai setiap volume yang mereka buat dengan tanda `costcenter=115` dan `stack=prod`. Jika pengguna tidak meneruskan tanda tertentu ini, atau jika mereka tidak menentukan tanda sama sekali, maka permintaan itu akan gagal.

Untuk tindakan-tindakan yang digunakan untuk membuat sumber daya yang menerapkan tanda, para pengguna juga harus memiliki izin untuk menggunakan tindakan `CreateTags`. Pernyataan kedua menggunakan kunci syarat `ec2:CreateAction` untuk memungkinkan para pengguna membuat tanda hanya dalam konteks `CreateVolume`. Para pengguna tidak dapat menandai volume yang ada atau sumber daya lainnya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCreateTaggedVolumes",
            "Effect": "Allow",
            "Action": "ec2:CreateVolume",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/costcenter": "115",
                    "aws:RequestTag/stack": "prod"
                }
            }
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateVolume"
                }
            }
        }
    ]
}
```

Kebijakan berikut memungkinkan para pengguna untuk membuat volume tanpa harus menentukan tanda. Tindakan `CreateTags` akan dievaluasi hanya jika tanda ditentukan dalam permintaan `CreateVolume`. Jika para pengguna menentukan tanda, maka tanda tersebut harus `purpose=test`. Tidak ada tanda lain yang diperbolehkan dalam permintaan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateVolume",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/purpose": "test",
                    "ec2:CreateAction": "CreateVolume"
                },
                "ForAllValues:StringEquals": {
```

```
                    "aws:TagKeys": "purpose"
                }
            }
        }
    ]
}
```

Contoh: Bekerja dengan volume menggunakan konsol Amazon EC2

Kebijakan berikut memberi pengguna izin untuk melihat dan membuat volume, serta melampirkan dan melepaskan volume ke instans tertentu menggunakan konsol Amazon EC2.

Para pengguna dapat melampirkan volume apa pun ke instans yang memiliki tanda "`purpose=test`" dan juga melepaskan volume yang dilampirkan dari instans tersebut. Untuk melampirkan volume menggunakan konsol Amazon EC2, akan sangat membantu bagi para pengguna jika mereka memiliki izin untuk menggunakan tindakan `ec2:DescribeInstances`, karena hal ini akan memungkinkan mereka memilih instans dari daftar yang sudah diisi sebelumnya dalam kotak dialog Lampirkan Volume. Akan tetapi, hal ini juga akan memungkinkan para pengguna untuk menampilkan semua instans dalam halaman Instans dalam konsol tersebut, sehingga Anda dapat menghilangkan tindakan ini.

Dalam pernyataan pertama, `ec2:DescribeAvailabilityZones` tindakan diperlukan untuk memastikan bahwa pengguna dapat memilih Zona Ketersediaan saat membuat volume.

Para pengguna tidak dapat memberikan tanda pada volume-volume yang mereka buat (baik ketika volume sedang dibuat maupun setelah volume dibuat).

Izinkan pengguna bekerja dengan snapshot

Berikut ini adalah contoh kebijakan-kebijakan untuk `CreateSnapshot` (snapshot titik per waktu dari volume EBS) dan `CreateSnapshots` (snapshot multi-volume).

Contoh

- Contoh: Membuat snapshot
- Contoh: Membuat beberapa snapshot
- Contoh: Membuat snapshot dengan tanda
- Contoh: Membuat snapshot multivolume dengan tanda
- Contoh: Menyalin beberapa snapshot

- Contoh: Memodifikasi pengaturan izin untuk snapshot

Contoh: Membuat snapshot

Kebijakan berikut memungkinkan pelanggan untuk menggunakan tindakan API CreateSnapshot. Pelanggan dapat membuat snapshot hanya jika volume sudah dienkripsi dan hanya jika ukuran volume kurang dari 20 GiB.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshot",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshot",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "NumericLessThan": {
                    "ec2:VolumeSize": "20"
                },
                "Bool": {
                    "ec2:Encrypted": "true"
                }
            }
        }
    ]
}
```

Contoh: Membuat beberapa snapshot

Kebijakan berikut memungkinkan pelanggan untuk menggunakan tindakan API CreateSnapshots. Pelanggan dapat membuat beberapa snapshot hanya jika semua volume pada instans memiliki tipe GP2.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshots",
         "Resource":[
"arn:aws:ec2:us-east-1::snapshot/*",
"arn:aws:ec2:*:*:instance/*"
   ]
      },
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshots",
         "Resource":"arn:aws:ec2:us-east-1:*:volume/*",
         "Condition":{
            "StringLikeIfExists":{
               "ec2:VolumeType":"gp2"
             }
     }

      }
   ]
}
```

Contoh: Membuat snapshot dengan tanda

Kebijakan berikut mencakup kunci syarat `aws:RequestTag` yang mengharuskan pelanggan menerapkan tanda `costcenter=115` dan `stack=prod` ke setiap snapshot baru. Jika pengguna tidak meneruskan tanda tertentu ini, atau jika mereka tidak menentukan tanda sama sekali, maka permintaan itu akan gagal.

Untuk tindakan-tindakan yang digunakan untuk membuat sumber daya yang menerapkan tanda, pelanggan juga harus memiliki izin untuk menggunakan tindakan `CreateTags`. Pernyataan ketiga menggunakan kunci syarat `ec2:CreateAction` untuk memungkinkan para pelanggan membuat tanda hanya dalam konteks `CreateSnapshot`. Para pelanggan tidak dapat menandai volume yang ada atau sumber daya lainnya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshot",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*"
        },
        {
            "Sid": "AllowCreateTaggedSnapshots",
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshot",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/costcenter": "115",
                    "aws:RequestTag/stack": "prod"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateSnapshot"
                }
            }
        }
    ]
}
```

Contoh: Membuat snapshot multivolume dengan tanda

Kebijakan berikut mencakup kunci syarat `aws:RequestTag` yang mengharuskan pelanggan menerapkan tanda `costcenter=115` dan `stack=prod` saat membuat set snapshot multivolume. Jika pengguna tidak meneruskan tanda tertentu ini, atau jika mereka tidak menentukan tanda sama sekali, maka permintaan itu akan gagal.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshots",
         "Resource":[
"arn:aws:ec2:us-east-1::snapshot/*",
"arn:aws:ec2:*:*:instance/*",
"arn:aws:ec2:*:*:volume/*"

   ]
      },
      {
         "Sid":"AllowCreateTaggedSnapshots",
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshots",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "aws:RequestTag/costcenter":"115",
               "aws:RequestTag/stack":"prod"
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":"ec2:CreateTags",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "ec2:CreateAction":"CreateSnapshots"
            }
         }
      }
   ]
}
```

Kebijakan berikut memungkinkan para pelanggan untuk membuat snapshot tanpa harus menentukan tanda. Tindakan `CreateTags` akan dievaluasi hanya jika tanda ditentukan dalam permintaan `CreateSnapshot` atau `CreateSnapshots`. Tanda dapat dihilangkan dalam permintaan. Jika tanda ditentukan, maka tanda tersebut harus `purpose=test`. Tidak ada tanda lain yang diperbolehkan dalam permintaan.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:CreateTags",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "aws:RequestTag/purpose":"test",
               "ec2:CreateAction":"CreateSnapshot"
            },
            "ForAllValues:StringEquals":{
               "aws:TagKeys":"purpose"
            }
         }
      }
   ]
}
```

Kebijakan berikut memungkinkan pelanggan untuk membuat set snapshot multivolume tanpa perlu menentukan tanda. Tindakan `CreateTags` akan dievaluasi hanya jika tanda ditentukan dalam permintaan `CreateSnapshot` atau `CreateSnapshots`. Tanda dapat dihilangkan dalam permintaan. Jika tanda ditentukan, maka tanda tersebut harus `purpose=test`. Tidak ada tanda lain yang diperbolehkan dalam permintaan.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshots",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:CreateTags",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "aws:RequestTag/purpose":"test",
               "ec2:CreateAction":"CreateSnapshots"
            },
            "ForAllValues:StringEquals":{
               "aws:TagKeys":"purpose"
            }
         }
      }
   ]
}
```

Kebijakan berikut mengizinkan snapshot untuk dibuat hanya jika volume sumber diberi tanda dengan `User:`*`username`* untuk pelanggan, dan snapshot itu sendiri diberi tanda dengan `Environment:Dev` dan `User:`*`username`*. Pelanggan dapat menambahkan tanda tambahan untuk snapshot tersebut.

Kebijakan untuk `CreateSnapshots` berikut mengizinkan snapshot untuk dibuat hanya jika volume sumber diberi tanda dengan `User:`*`username`* untuk pelanggan, dan snapshot itu sendiri diberi tanda dengan `Environment:Dev` dan `User:`*`username`*.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": "arn:aws:ec2:us-east-1:*:instance/*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/User": "${aws:username}"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Environment": "Dev",
                    "aws:RequestTag/User": "${aws:username}"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*"
        }
    ]
}
```

Kebijakan berikut memungkinkan penghapusan snapshot hanya jika snapshot tersebut diberi tanda dengan Pengguna:username untuk pelanggan.

JSON

```
{
   "Version":"2012-10-17",
```

```
   "Statement": [
            {
         "Effect":"Allow",
         "Action":"ec2:DeleteSnapshot",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "aws:ResourceTag/User":"${aws:username}"
            }
         }
      }
   ]
}
```

Kebijakan berikut memungkinkan pelanggan untuk membuat snapshot tetapi menolak tindakan jika snapshot yang dibuat memiliki kunci tanda `value=stack`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":[
            "ec2:CreateSnapshot",
            "ec2:CreateTags"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Deny",
         "Action":"ec2:CreateSnapshot",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "ForAnyValue:StringEquals":{
               "aws:TagKeys":"stack"
            }
         }
      }
   ]
```

```
}
```

Kebijakan berikut memungkinkan pelanggan untuk membuat beberapa snapshot tetapi menolak tindakan jika snapshot yang dibuat tersebut memiliki kunci tanda `value=stack`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect":"Allow",
         "Action":[
            "ec2:CreateSnapshots",
            "ec2:CreateTags"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Deny",
         "Action":"ec2:CreateSnapshots",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "ForAnyValue:StringEquals":{
               "aws:TagKeys":"stack"
            }
         }
      }
   ]
}
```

Kebijakan berikut memungkinkan Anda untuk menggabungkan beberapa tindakan ke dalam satu kebijakan. Anda hanya dapat membuat snapshot (dalam konteks `CreateSnapshots`) saat snapshot tersebut dibuat di Wilayah `us-east-1`. Anda hanya dapat membuat beberapa snapshot (dalam konteks `CreateSnapshots`) ketika snapshot-snapshot tersebut sedang dibuat di Wilayah `us-east-1` dan ketika tipe instans-nya adalah `t2*`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
          {
         "Effect":"Allow",
         "Action":[
            "ec2:CreateSnapshots",
            "ec2:CreateSnapshot",
            "ec2:CreateTags"
         ],
         "Resource": [
            "arn:aws:ec2:*:*:instance/*",
            "arn:aws:ec2:*:*:snapshot/*",
            "arn:aws:ec2:*:*:volume/*"
         ],
         "Condition":{
            "StringEqualsIgnoreCase": {
              "ec2:Region": "us-east-1"
            },
            "StringLikeIfExists": {
              "ec2:InstanceType": ["t2.*"]
            }
         }
      }
   ]
}
```

Contoh: Menyalin beberapa snapshot

Resource-level izin yang ditentukan untuk CopySnapshot tindakan berlaku untuk snapshot baru dan snapshot sumber.

Kebijakan contoh berikut mengizinkan prinsipal utama untuk menyalin snapshot hanya jika snapshot baru dibuat dengan tanda kunci `purpose` dan nilai tanda `production` (`purpose=production`).

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCopySnapshotWithTags",
            "Effect": "Allow",
            "Action": "ec2:CopySnapshot",
            "Resource": "arn:aws:ec2:*:111122223333:snapshot/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/purpose": "production"
                }
            }
        }
    ]
}
```

Kebijakan contoh berikut mengizinkan kepala sekolah untuk menyalin snapshot hanya jika snapshot sumber dimiliki oleh AWS akun. `123456789012`

Kebijakan contoh berikut mengizinkan prinsipal untuk menyalin snapshot hanya jika snapshot sumber dibuat dengan kunci tag of. `CSISnapshotName`

```
{
    "Effect": "Allow",
    "Action": "ec2:CopySnapshot",
    "Resource": "arn:aws:ec2:*::snapshot/${*}",
    "Condition": {
        "StringLike": {
           "aws:RequestTag/CSISnapshotName": "*"
         }
     }
},

{
    "Effect": "Allow",
    "Action": "ec2:CopySnapshot",
    "Resource": "arn:aws:ec2:*::snapshot/snap-*"
}
```

Contoh: Memodifikasi pengaturan izin untuk snapshot

Kebijakan berikut mengizinkan modifikasi snapshot hanya jika snapshot diberi tag`User:`*`username`*, di *`username`* mana nama pengguna AWS akun pelanggan. Permintaan akan gagal jika syarat ini tidak dipenuhi.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
           {
         "Effect":"Allow",
         "Action":"ec2:ModifySnapshotAttribute",
         "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
         "Condition":{
            "StringEquals":{
               "aws:ResourceTag/user-name":"${aws:username}"
            }
         }
      }
   ]
}
```

Memecahkan masalah otorisasi Amazon EBS

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan Amazon EBS dan IAM.

Masalah

- Saya tidak berwenang untuk melakukan tindakan di Amazon EBS
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang-orang di luar saya Akun AWS untuk mengakses sumber daya Amazon EBS saya

Saya tidak berwenang untuk melakukan tindakan di Amazon EBS

Jika member Konsol Manajemen AWS itahu Anda bahwa Anda tidak berwenang untuk melakukan tindakan, maka Anda harus menghubungi administrator untuk mendapatkan bantuan. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Contoh kesalahan berikut terjadi ketika pengguna `mateojackson` IAM mencoba menggunakan konsol untuk melihat detail tentang volume tetapi tidak memiliki `ec2:DescribeVolumes` izin.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ec2:DescribeVolumes on resource: volume-id
```

Dalam hal ini, Mateo meminta AWS administratornya untuk mengizinkannya menggambarkan volume.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan bahwa Anda tidak berwenang untuk melakukan `iam:PassRole` tindakan tersebut, kebijakan Anda harus diperbarui untuk memungkinkan Anda meneruskan peran ke Amazon EBS.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan itu alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di Amazon EBS. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang-orang di luar saya Akun AWS untuk mengakses sumber daya Amazon EBS saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACL), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mengetahui apakah Amazon EBS mendukung fitur-fitur ini, lihatCara kerja Amazon EBS dengan IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM lainnya Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyedi Akun AWS akan akses milik pihak ketiga di Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Validasi kepatuhan untuk Amazon EBS

Pengalaman AWS pendaftaran baru kami tidak dirancang untuk beban kerja yang diatur. Jika Anda menggunakan pengalaman AWS pendaftaran baru kami, tetapi ingin menggunakannya AWS untuk beban kerja yang diatur, Anda dapat mendaftar untuk AWS (lanjutan) atau mengaktifkan fitur lanjutan untuk lingkungan Anda AWS .

Untuk mengetahui apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup berdasarkan Program Kepatuhan dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program Kep AWS atuhan Program Jaminan .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

Ketahanan data di Amazon EBS

Infrastruktur AWS global dibangun di sekitar Wilayah AWS dan Availability Zones. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang 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.

Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat Infrastruktur AWS Global.

Selain infrastruktur AWS global, Amazon EBS menawarkan beberapa fitur untuk membantu mendukung ketahanan data dan kebutuhan pencadangan Anda.

- Menerapkan otomatisasi snapshot EBS menggunakan Amazon Data Lifecycle Manager
- Menyalin snapshot EBS di seluruh Wilayah

Alat pemantauan untuk Amazon EBS

Pemantauan merupakan bagian penting dalam menjaga keandalan, ketersediaan, dan kinerja Amazon Elastic Block Store dan AWS solusi Anda yang lain. AWS menyediakan alat pemantauan berikut untuk menonton Amazon EBS, melaporkan ketika ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu:

- AWS CloudTrailmenangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama Anda Akun AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang dipanggil AWS, alamat IP sumber dari mana panggilan dilakukan, dan kapan panggilan terjadi. API untuk mengelola volume dan snapshot EBS Anda adalah bagian dari Amazon EC2 API. Untuk informasi selengkapnya tentang CloudTrail dan API Amazon EC2, lihat Log panggilan API Amazon EC2 AWS CloudTrail menggunakan dalam Panduan Pengguna Amazon EC2.
- Amazon CloudWatch memantau AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS 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. Misalnya, Anda dapat CloudWatch melacak penggunaan CPU atau metrik lain dari instans Amazon EC2 Anda dan secara otomatis meluncurkan instans baru bila diperlukan. Untuk informasi selengkapnya, lihat the section called “Amazon CloudWatch”.
- Amazon EventBridge dapat digunakan untuk mengotomatiskan AWS layanan Anda dan merespons secara otomatis peristiwa sistem, seperti masalah ketersediaan aplikasi atau perubahan sumber daya. Acara dari AWS layanan dikirimkan ke EventBridge dalam waktu dekat. Anda dapat menuliskan aturan sederhana untuk menunjukkan peristiwa mana yang sesuai kepentingan Anda, dan tindakan otomatis mana yang diambil ketika suatu peristiwa sesuai dengan suatu aturan. Untuk informasi selengkapnya, lihat the section called “Amazon EventBridge”.
- Statistik kinerja terperinci Amazon EBS menyediakan statistik I/O kinerja real-time untuk volume Amazon EBS yang dilampirkan ke instans Amazon Nitro-based EC2. Untuk informasi selengkapnya, lihat Statistik kinerja terperinci Amazon EBS.
- Amazon GuardDuty membantu mendeteksi aktivitas yang berpotensi berbahaya di instans EC2 Anda. GuardDuty Perlindungan Malware untuk EC2 memindai volume EBS yang dilampirkan ke instans EC2 Anda. Untuk informasi selengkapnya, lihat the section called “Amazon GuardDuty”.

CloudWatch Metrik Amazon untuk Amazon EBS

CloudWatch Metrik Amazon adalah data statistik yang dapat Anda gunakan untuk melihat, menganalisis, dan mengatur alarm pada perilaku operasional volume Anda.

Data tersedia secara otomatis dalam periode 1 menit tanpa biaya.

Saat Anda mendapatkan data dari CloudWatch, Anda dapat menyertakan parameter `Period` permintaan untuk menentukan perincian data yang dikembalikan. Ini berbeda dengan periode yang digunakan EBS untuk mengumpulkan data (periode 1 menit). Kami menyarankan Anda menentukan periode dalam permintaan Anda yang sama dengan atau lebih besar dari periode pengumpulan untuk memastikan bahwa data yang dikembalikan valid.

Anda bisa mendapatkan data menggunakan CloudWatch API atau konsol Amazon EC2. Konsol mengambil data mentah dari CloudWatch API dan menampilkan serangkaian grafik berdasarkan data. Bergantung pada kebutuhan Anda, Anda mungkin lebih memilih menggunakan data dari API atau grafik di konsol.

Topik

- Metrik untuk volume Amazon EBS
- Metrik untuk snapshot Amazon EBS
- Metrik untuk instans Nitro
- Metrik untuk pemulihan snapshot cepat
- Grafik konsol Amazon EC2

Metrik untuk volume Amazon EBS

Namespace `AWS/EBS` mencakup metrik berikut untuk volume EBS yang dilampirkan ke semua tipe instans. Semua jenis volume Amazon EBS secara otomatis mengirim metrik 1 menit ke CloudWatch, tetapi hanya jika volume dilampirkan ke instans.

Untuk mendapatkan informasi tentang ruang disk yang tersedia dari sistem operasi di instans, lihat Lihat ruang disk kosong.

> **Note**
>
> Beberapa metrik memiliki perbedaan pada instans yang dibuat pada Sistem Nitro. Untuk daftar jenis instance ini, lihat Instans yang dibangun di Sistem Nitro.

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeAvgIOPS *`	**Note** Didukung untuk semua jenis volume EBS yang terpasang ke instans Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Rata-rata IOPS baca dan tulis didorong ke volume dalam satu menit. Jika tidak ada operasi yang didorong ke volume dalam menit terakhir, maka nilai untuk metrik adalah nol (`0`). Untuk informasi selengkapnya, lihat Memantau I/O	Ops/s	`VolumeId` | `InstanceId`	• `Sum` • `Average` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	karakteristik menggunakan CloudWatch. Untuk volume yang Multi-Attach diaktifkan, gunakan `InstanceId` dimensi untuk melihat IOPS rata-rata untuk lampiran instans volume tertentu.			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeAvgThroughput *`	**Note** Didukung untuk semua jenis volume EBS yang terpasang ke instans Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Rata-rata throughput baca dan tulis didorong ke volume dalam satu menit. Jika tidak ada operasi yang didorong ke volume dalam menit terakhir, maka nilai untuk metrik adalah nol (`0`). Untuk informasi selengkapnya, lihat Memantau I/O karakteristik menggunakan CloudWatch. Untuk volume yang Multi-Attach diaktifkan,	KiB/s	`VolumeId` | `InstanceId`	• `Sum` • `Average` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	gunakan `InstanceId` dimensi untuk melihat throughput rata-rata untuk lampiran instans volume tertentu..			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeAvgReadLatency *`	**Note** Didukung untuk semua jenis volume yang dilampirkan ke instance Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Rata-rata waktu yang dibutuhkan untuk menyelesaikan operasi baca dalam satu menit. Gunakan metrik ini untuk memantau I/O latensi rata-rata volume EBS yang terpasang ke instans Amazon EC2 Anda. Rata-rata dihitung berdasarkan I/O operasi yang diselesaikan pada menit terakhir. Jika tidak ada operasi yang diselesaikan dalam menit	Milidetik	`VolumeId` | `InstanceId`	`Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	terakhir, maka nilai untuk metrik adalah nol. Untuk volume yang Multi-Attach diaktifkan, gunakan `InstanceId` dimensi untuk melihat latensi rata-rata untuk lampiran instans volume tertentu.			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeAvgWriteLatency *`	**Note** Didukung untuk semua jenis volume yang dilampirkan ke instance Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Rata-rata waktu yang dibutuhkan untuk menyelesaikan operasi penulisan dalam satu menit. Gunakan metrik ini untuk memantau I/O latensi rata-rata volume EBS yang terpasang ke instans Amazon EC2 Anda. Rata-rata dihitung berdasarkan I/O operasi yang diselesaikan pada menit terakhir. Jika tidak ada operasi yang diselesaikan dalam menit	Milidetik	`VolumeId` | `InstanceId`	`Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	terakhir, maka nilai untuk metrik adalah nol. Untuk volume yang Multi-Attach diaktifkan, gunakan `InstanceId` dimensi untuk melihat latensi rata-rata untuk lampiran instans volume tertentu.			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeIOPSExceededCheck *`	**Note** Didukung untuk semua jenis volume, kecuali magnetic (`standard`), yang dilampirkan ke instance Nitro. Tidak didukung dengan volume Multi-Attach yang diaktifkan. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Melaporkan apakah aplikasi secara konsisten berusaha mendorong IOPS yang melebihi kinerja IOPS yang disediakan volume dalam menit terakhir. Metrik ini dapat berupa `0` (IOPS yang disediakan tidak terlampaui) atau `1` (IOPS yang disediaka	Tidak ada	`VolumeId` | `InstanceId`	• `Sum` • `Average` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	n terlampaui). Untuk informasi selengkap nya, lihat Memantau I/O karakteristik menggunak an CloudWatch.			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeThroughputExceededCheck *`	**Note** Didukung untuk semua jenis volume, kecuali magnetic (`standard`), yang dilampirkan ke instance Nitro. Tidak didukung dengan volume Multi-Attach yang diaktifkan. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. Melaporkan apakah aplikasi secara konsisten berusaha mendorong throughput yang melebihi kinerja throughput yang disediakan volume dalam menit terakhir. Metrik ini dapat berupa `0` (throughput yang disediakan tidak terlampaui) atau 1 (throughput yang	Tidak ada	`VolumeId` | `InstanceId`	• `Sum` • `Average` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	disediakan terlampaui) .Untuk informasi selengkapnya, lihat. Memantau I/O karakteristik menggunakan CloudWatch			

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeReadBytes`	Memberikan informasi tentang operasi yang dibaca dalam periode waktu tertentu. • Statistik `Sum` melaporkan jumlah total byte yang ditransfer selama periode tersebut. • `Average` Statistik melaporkan jumlah rata-rata byte yang dibaca selama periode yang ditentukan. • `SampleCount` Statistik mewakili jumlah titik data yang digunakan dalam perhitungan statistik. **Note** Untuk instans Xen, data dilaporkan hanya ketika ada aktivitas baca pada volume.	Byte	`VolumeId`	• `Average` • `Sum` • `SampleCount` • `Minimum` | `Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeWriteBytes`	Memberikan informasi tentang operasi baca dalam jangka waktu tertentu • Statistik `Sum` melaporkan jumlah total bita yang ditransfer selama periode tersebut. • `Average`Statistik melaporkan jumlah rata-rata byte yang ditulis selama periode yang ditentukan. • `SampleCount` Statistik mewakili jumlah titik data yang digunakan dalam perhitungan statistik. **Note** Untuk instans Xen, data dilaporkan hanya ketika ada aktivitas baca pada volume.	Byte	`VolumeId`	• `Average` • `Sum` • `SampleCount` • `Minimum`| `Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeReadOps`	Total jumlah operasi baca dalam periode waktu tertentu. Operasi baca dihitung setelah selesai. Untuk menghitung operasi baca rata-rata per detik (baca IOPS) untuk periode, bagi total operasi baca dalam periode tersebut dengan jumlah detik dalam periode tersebut.	Hitungan	`VolumeId`	• `Average` • `Sum` • `Minimum`|`Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance
`VolumeWriteOps`	Total jumlah operasidalam periode waktu tertentu. Operasi menulis dihitung setelah selesai. Untuk menghitung rata-rata operasi tulis per detik (IOPS tulis) untuk periode tersebut, bagi total operasi tulis dalam periode dengan jumlah detik dalam periode tersebut.	Hitungan	`VolumeId`	• `Average` • `Sum` • `Minimum`|`Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeTotalReadTime`	**Note** Tidak didukung dengan volume Multi-Attach yang diaktifkan. Untuk instans Xen, data dilaporkan hanya ketika ada aktivitas baca pada volume. Total jumlah detik yang dihabiskan oleh semua operasi baca yang selesai dalam jangka waktu tertentu. Jika beberapa permintaan dikirimkan pada waktu yang sama, total ini dapat lebih besar dari lama periode. Misalnya, selama 1 menit (60 detik): jika 150 operasi selesai selama periode tersebut, dan setiap operasi memerlukan 1 detik, nilainya adalah 150 detik.	Detik	`VolumeId`	• `Average` – tidak relevan untuk volume yang dilampirkan ke Nitro-based instance • `Sum` • `Minimum`|`Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeTotalWriteTime`	**Note** Tidak didukung dengan volume Multi-Attach yang diaktifkan. Untuk instans Xen, data dilaporkan hanya ketika ada aktivitas baca pada volume. Total jumlah detik yang dihabiskan oleh semua operasiyang selesai dalam periode waktu tertentu. Jika beberapa permintaan dikirimkan pada waktu yang sama, total ini dapat lebih besar dari lama periode. Misalnya, selama 1 menit (60 detik): jika 150 operasi selesai selama periode tersebut, dan setiap operasi memerlukan 1 detik, nilainya adalah 150 detik.	Detik	`VolumeId`	• `Average`- tidak relevan untuk volume yang dilampirkan ke Nitro-based instance • `Sum` • `Minimum`| `Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeIdleTime`	**Note** Tidak didukung dengan volume Multi-Attach yang diaktifkan. Total jumlah detik dalam periode waktu tertentu ketika tidak ada operasi baca atauyang .	Detik	`VolumeId`	• `Average`- tidak relevan untuk volume yang dilampirkan ke Nitro-based instance • `Sum` • `Minimum`| `Maximum` — hanya untuk volume yang dilampirkan ke Nitro-based instance

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeQueueLength`	Jumlah permintaan operasi baca danharus diselesaikan dalam jangka waktu tertentu.	Hitungan	`VolumeId`	• `Average` • `Sum` — tidak relevan untuk volume yang dilampirkan ke instans Nitro • `Minimum` | `Maximum` — hanya untuk volume yang dilampirkan ke instans Nitro

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeStalledIOCheck *`	**Note** Hanya untuk contoh Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke tugas Amazon ECS dan AWS Fargate . Melaporkan apakah volume telah melewati atau gagal pemeriksaan IO yang terhenti di menit terakhir.Metrik ini dapat berupa `0` (lulus) atau 1 (gagal). Untuk informasi selengkapnya, lihat Memantau I/O karakteristik menggunakan CloudWatch.	Tidak ada	`VolumeId` | `InstanceId`	• Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`VolumeThroughputPercentage`	**Note** Khusus volume SSD IOPS yang tersedia. Tidak didukung dengan volume Multi-Attach yang diaktifkan. Persentase I/O operasi per detik (IOPS) yang dikirimkan dari total IOPS yang disediakan untuk volume Amazon EBS. Volume SSD IOPS yang Tersedia memberikan performa yang telah disediakan 99,9 persen. Selama penulisan, jika tidak ada I/O permintaan tertunda lainnya dalam satu menit, nilai metrik akan menjadi 100 persen. Selain itu, I/O kinerja volume dapat menurun sementara karena tindakan yang telah Anda lakukan (misalnya, membuat snapshot volume selama	Persen	`VolumeId`	• `Average` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
	penggunaan puncak, menjalankan volume pada non- EBS-optimized instans, atau mengakses data pada volume untuk pertama kalinya).			
`VolumeConsumedReadWriteOps`	**Note** Khusus volume SSD IOPS yang tersedia. Jumlah total operasi baca dan tulis (dinormalisasi ke unit kapasitas 256K) yang dikonsumsi dalam periode waktu tertentu. I/O operasi yang lebih kecil dari 256K masing-masing dihitung sebagai 1 IOPS yang dikonsumsi. I/O operasi yang lebih besar dari 256K dihitung dalam unit kapasitas 256K. Misalnya, 1024K I/O akan dihitung sebagai 4 IOPS yang dikonsumsi.	Hitungan	`VolumeId`	• `Average` • `Sum` • `Minimum` | `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`BurstBalance`	**Note** `gp2`,`st1`, dan `sc1` volume saja. Menyediakan informasi tentang persentase I/O kredit (untuk`gp2`) atau kredit throughput (untuk `st1` dan`sc1`) yang tersisa di bucket burst. Data dilaporkan CloudWatch hanya ketika volume aktif. Jika volume tidak dipasang, tidak ada data yang dilaporkan. Jika performa dasar volume melebihi performa lonjakan maksimum, kredit tidak pernah dihabiskan. Untuk instans lainnya, saldo lonjakan yang dilaporkan adalah 100%. Untuk informasi selengkapnya, lihat `kinerja volume gp2`.	Persen	`VolumeId`	• `Average` • `Sum` — tidak relevan untuk volume yang dilampirkan ke instans Nitro. • `Minimum` | `Maximum`

* Anda tidak dapat menggunakan metrik ini di Pos Terdepan, Zona Lokal, atau Zona Panjang Gelombang.

Metrik untuk snapshot Amazon EBS

`AWS/EBS`Namespace menyertakan metrik berikut untuk snapshot Amazon EBS.

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`SnapshotCopyBytesTransferred`	Jumlah data snapshot yang disalin ke suatu Wil AWS ayah.	Byte	`sourceRegion`	`Sum`

Metrik untuk instans Nitro

`AWS/EC2`Namespace menyertakan metrik Amazon EBS tambahan untuk volume yang dilampirkan ke Nitro-based instans yang bukan instans bare metal.

Metrik	Deskripsi	Unit	Statistik yang bermakna
`InstanceEBSIOPSExceededCheck *`	Melaporkan apakah aplikasi mencoba mendorong IOPS yang melebihi batas EBS IOPS maksimum untuk instans dalam menit terakhir. Metrik ini dapat berupa `0` (IOPS tidak terlampaui) atau 1 (IOPS terlampaui).	Tidak ada	• Jumlah • Rata-rata • Minimum • Maksimum
`InstanceEBSThroughputExceededCheck *`	Melaporkan apakah aplikasi mencoba mendorong throughput yang melebihi batas throughput EBS maksimum untuk instans dalam menit terakhir. Metrik ini dapat berupa `0` (throughput tidak terlampaui) atau 1 (throughput terlampaui).	Tidak ada	• Jumlah • Rata-rata • Minimum • Maksimum
`EBSReadOps`	Operasi baca yang diselesaikan dari semua volume Amazon EBS yang dilampirkan ke instans dalam periode waktu yang ditentuka	Hitungan	• Jumlah • Rata-rata

Metrik	Deskripsi	Unit	Statistik yang bermakna
	n. Untuk menghitung I/O operasi baca rata-rata per detik (Baca IOPS) untuk periode tersebut, bagi total operasi dalam periode dengan jumlah detik dalam periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi jumlah ini dengan 300 untuk menghitung nilai IOPS Baca. Jika Anda memiliki pemantauan terperinc i (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan operasi per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSReadOps m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembal ikan metrik operations/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.		• Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
`EBSWriteOps`	Operasi tulis yang diselesaikan ke semua volume EBS yang dilampirkan pada instans tersebut dalam periode waktu yang ditentukan. Untuk menghitung I/O operasi tulis rata-rata per detik (Tulis IOPS) untuk periode tersebut, bagi total operasi dalam periode dengan jumlah detik dalam periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menghitung nilai IOPS Tulis. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan operasi per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSWriteOps m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik operations/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.	Hitungan	• Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
EBSReadBytes	Bita yang dibaca dari semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu. Jumlah yang dilaporkan adalah jumlah bita baca selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Baca Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik DIFF_TIME untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai EBSReadBytes m1, rumus matematika metrik m1/(DIFF_TIME(m1)) mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang DIFF_TIME dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.	Byte	• Jumlah • Rata-rata • Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
`EBSWriteBytes`	Bita yang ditulis ke semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu. Jumlah yang dilaporkan adalah jumlah bita tulis selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Tulis Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSWriteBytes m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.	Byte	• Jumlah • Rata-rata • Minimum • Maksimum
`EBSIOBalance%`	Memberikan informasi tentang persentase I/O kredit yang tersisa di bucket burst. Metrik ini hanya tersedia untuk pemantauan dasar. Metrik ini hanya tersedia untuk beberapa ukuran instans `*.4xlarge` dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali. Untuk informasi selengkapnya, lihat EBS dioptimalkan secara default. Statistik `Sum` tidak berlaku untuk metrik ini.	Persen	• Minimum • Maksimum

Metrik	Deskripsi	Unit	Statistik yang bermakna
`EBSByteBalance%`	Memberikan informasi tentang persentase kredit throughput yang tersisa di bucket lonjakan. Metrik ini hanya tersedia untuk pemantauan dasar. Metrik ini hanya tersedia untuk beberapa ukuran instans `*.4xlarge` dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali. Untuk informasi selengkapnya, lihat EBS dioptimalkan secara default. Statistik `Sum` tidak berlaku untuk metrik ini.	Persen	• Minimum • Maksimum

* Anda tidak dapat menggunakan metrik ini di Pos Terdepan, Zona Lokal, atau Zona Panjang Gelombang.

Metrik untuk pemulihan snapshot cepat

Namespace `AWS/EBS` menyertakan metrik untuk pemulihan snapshot cepat.

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
`FastSnapshotRestoreCreditsBucketSize`	Jumlah maksimal volume akan membuatyang dapat diakumulasi. Metrik ini dilaporkan per snapshot per Availability Zone.	Tidak ada	`SnapshotId` | `AvailabilityZone`	• `Average` • `Minimum` | `Maximum` **Note** Statistik yang paling bermakna adalah `Average`. Hasil untuk statistik `Minimum` dan `Maximum`

Metrik	Deskripsi	Unit	Dimensi	Statistik yang bermakna
				adalah sama dengan `Average` dan dapat digunakan.
`FastSnapshotRestoreCreditsBalance`	Jumlah volume membuattersedia. Metrik ini dilaporkan per snapshot per Zona Ketersediaan.	Tidak ada	`SnapshotId | AvailabilityZone`	• `Average` • `Minimum` | `Maximum` **Note** Statistik yang paling bermakna adalah `Average`. Hasil untuk statistik `Minimum` dan `Maximum` adalah sama dengan `Average` dan dapat digunakan.

Grafik konsol Amazon EC2

Setelah Anda membuat volume, Anda dapat melihat grafikvolume di konsol Amazon EC2. Pilih volume pada Volume di konsol dan pilih Pemantauan. Tabel berikut mencantumkan grafik yang ditampilkan. Kolom di sebelah kanan menjelaskan bagaimana metrik data mentah dari CloudWatch API digunakan untuk menghasilkan setiap grafik. Periode untuk semua grafik adalah 5 menit.

Grafik	Deskripsi menggunakan metrik mentah
Baca throughput (KiB/s)	`Sum(VolumeReadBytes) / Period / 1024`
Tulis throughput (KiB/s)	`Sum(VolumeWriteBytes) / Period / 1024`

Grafik	Deskripsi menggunakan metrik mentah
Baca operasi (Ops/s)	`Sum(VolumeReadOps) / Period`
Menulis operasi (Ops/s)	`Sum(VolumeWriteOps) / Period`
Panjang antrean rata-rata (Operasi)	`Avg(VolumeQueueLength)`
Waktu yang dihabiskan untuk idle (%)	`Sum(VolumeIdleTime) / Period × 100`

Ukuran baca rata-rata (KiB/op)	`Avg(VolumeReadBytes) / 1024`
	
	Misalnya Nitro-based , rumus berikut memperoleh Ukuran Baca Rata-rata menggunakan CloudWatch Metrik Matematika:
	
	`(Sum(VolumeReadBytes) / Sum(Volum eReadOps)) / 1024`
	
	`VolumeReadOps` Metrik `VolumeReadBytes` dan tersedia di CloudWatch konsol EBS.

Ukuran tulis rata-rata (KiB/op)	`Avg(VolumeWriteBytes) / 1024`
	
	Misalnya, Nitro-based rumus berikut memperoleh Ukuran Tulis Rata-Rata menggunakan CloudWatch Metrik Matematika:
	
	`(Sum(VolumeWriteBytes) / Sum(Volum eWriteOps)) / 1024`
	
	`VolumeWriteOps` Metrik `VolumeWriteBytes` dan tersedia di CloudWatch konsol EBS.

Grafik	Deskripsi menggunakan metrik mentah

Latensi baca rata-rata (ms/op)	`Avg(VolumeTotalReadTime) × 1000`
	
	Misalnya, Nitro-based rumus berikut memperoleh Rata-rata Read Latency menggunakan CloudWatch Metric Math:
	
	`(Sum(VolumeTotalReadTime) / Sum(VolumeReadOps)) × 1000`
	
	`VolumeReadOps` Metrik `VolumeTotalReadTime` dan tersedia di CloudWatch konsol EBS.

Latensi tulis rata-rata (ms/op)	`Avg(VolumeTotalWriteTime) × 1000`
	
	Sebagai Nitro-based contoh, rumus berikut memperoleh Rata-rata Write Latency menggunakan CloudWatch Metric Math:
	
	`(Sum(VolumeTotalWriteTime) / Sum(VolumeWriteOps)) * 1000`
	
	`VolumeWriteOps` Metrik `VolumeTotalWriteTime` dan tersedia di CloudWatch konsol EBS.

Untuk grafik latensi rata-rata dan grafik ukuran rata-rata, rata-ratanya dihitung terhadap jumlah total operasi (baca atau tulis, mana saja yang berlaku untuk grafik) yang selesai selama periode.

Acara EventBridge Amazon untuk Amazon EBS

Amazon EBS mengirimkan acara ke Amazon EventBridge untuk tindakan yang dilakukan pada volume dan snapshot. Dengan EventBridge, Anda dapat menetapkan aturan yang memicu tindakan terprogram sebagai respons terhadap peristiwa ini. Misalnya, Anda dapat membuat aturan yang mengirimkan notifikasi ke email Anda saat snapshot diaktifkan untuk pemulihan snapshot cepat.

Peristiwa di EventBridge direpresentasikan sebagai objek JSON. Kolom-kolom yang unik untuk peristiwa tersebut terdapat di bagian "detail" dari objek JSON. Bidang "peristiwa" berisi nama peristiwa. Bidang "hasil" berisi status selesai dari tindakan yang memicu peristiwa. Untuk informasi

selengkapnya, lihat pola EventBridge peristiwa Amazon di Panduan EventBridge Pengguna Amazon.

Untuk informasi lebih lanjut, lihat Apa itu Amazon EventBridge? dalam Panduan EventBridge Pengguna Amazon.

Peristiwa

- Peristiwa volume EBS
- Acara inisialisasi volume EBS
- Peristiwa modifikasi volume EBS
- Peristiwa snapshot EBS
- Peristiwa Arsip Snapshots EBS
- Peristiwa pemulihan snapshot cepat EBS
- Penggunaan AWS Lambda untuk menangani EventBridge peristiwa

Peristiwa volume EBS

Amazon EBS mengirimkan peristiwa ke EventBridge saat peristiwa volume berikut terjadi.

Peristiwa

- Buat volume (buatVolume)
- Hapus volume (hapusVolume)
- Pasang atau pasangkan ulang volume (attachVolume, reattachVolume)
- Lepaskan volume (DetachVolume)

Buat volume (buatVolume)

A `createVolume` cara dikirim ke AWS akun Anda ketika tindakan untuk membuat volume selesai. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Peristiwa ini dapat membawa hasil `available` atau `failed`. Pembuatan akan gagal jika tidak valid AWS KMS key diberikan, seperti yang ditunjukkan pada contoh di bawah ini.

Data peristiwa

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS untuk peristiwa `createVolume` yang berhasil.

```
{
   "version": "0",
   "id": "01234567-0123-0123-0123-012345678901",
   "detail-type": "EBS Volume Notification",
   "source": "aws.ec2",
   "account": "012345678901",
   "time": "yyyy-mm-ddThh:mm:ssZ",
   "region": "us-east-1",
   "resources": [
      "arn:aws:ec2:us-east-1:012345678901:volume/vol-01234567"
   ],
   "detail": {
      "result": "available",
      "cause": "",
      "event": "createVolume",
      "request-id": "01234567-0123-0123-0123-0123456789ab"
   }
}
```

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `createVolume` yang gagal. Penyebab kegagalan tersebut adalah kunci KMS yang dinonaktifkan.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "EBS Volume Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "sa-east-1",
  "resources": [
    "arn:aws:ec2:sa-east-1:0123456789ab:volume/vol-01234567",
  ],
  "detail": {
    "event": "createVolume",
    "result": "failed",
    "cause": "arn:aws:kms:sa-
east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab is disabled.",
    "request-id": "01234567-0123-0123-0123-0123456789ab",
  }
}
```

Berikut adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `createVolume` yang gagal. Penyebab kegagalan adalah impor utama KMS yang tertunda.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "EBS Volume Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "sa-east-1",
  "resources": [
    "arn:aws:ec2:sa-east-1:0123456789ab:volume/vol-01234567",
  ],
  "detail": {
    "event": "createVolume",
    "result": "failed",
    "cause": "arn:aws:kms:sa-
east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab is pending import.",
    "request-id": "01234567-0123-0123-0123-0123456789ab",
  }
}
```

Hapus volume (hapusVolume)

A `deleteVolume` cara dikirim ke AWS akun Anda ketika tindakan untuk menghapus volume selesai. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Peristiwa ini memiliki hasil `deleted`. Jika penghapusan tidak selesai, peristiwa tidak akan pernah dikirim.

Data peristiwa

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS untuk peristiwa `deleteVolume` yang berhasil.

```
{
   "version": "0",
   "id": "01234567-0123-0123-0123-012345678901",
   "detail-type": "EBS Volume Notification",
   "source": "aws.ec2",
   "account": "012345678901",
   "time": "yyyy-mm-ddThh:mm:ssZ",
   "region": "us-east-1",
   "resources": [
```

```
      "arn:aws:ec2:us-east-1:012345678901:volume/vol-01234567"
   ],
   "detail": {
      "result": "deleted",
      "cause": "",
      "event": "deleteVolume",
      "request-id": "01234567-0123-0123-0123-0123456789ab"
   }
}
```

Pasang atau pasangkan ulang volume (attachVolume, reattachVolume)

A `attachVolume` `reattachVolume` cara atau dikirim ke AWS akun Anda saat volume dilampirkan atau dilampirkan kembali ke instans. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Jika Anda menggunakan kunci KMS untuk mengenkripsi volume EBS dan kunci KMS tersebut menjadi tidak valid, EBS akan membuat catatan peristiwa jika kunci KMS tersebut kemudian digunakan untuk memasang atau memasang kembali instans, seperti yang ditunjukkan pada instans di bawah ini.

Data peristiwa

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `attachVolume` yang gagal. Penyebab kegagalan adalah penghapusan kunci KMS yang tertunda.

> **Note**
>
> AWS mungkin mencoba untuk memasang kembali ke volume setelah pemeliharaan server rutin.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "EBS Volume Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
  "arn:aws:ec2:us-east-1:0123456789ab:volume/vol-01234567",
  "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab"
  ],
  "detail": {
```

```
    "event": "attachVolume",
    "result": "failed",
    "cause": "arn:aws:kms:us-
east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab is pending deletion.",
    "request-id": ""
  }
}
```

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `reattachVolume` yang gagal. Penyebab kegagalan adalah penghapusan kunci KMS yang tertunda.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "EBS Volume Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
  "arn:aws:ec2:us-east-1:0123456789ab:volume/vol-01234567",
  "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab"
  ],
  "detail": {
    "event": "reattachVolume",
    "result": "failed",
    "cause": "arn:aws:kms:us-
east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab is pending deletion.",
    "request-id": ""
  }
}
```

Lepaskan volume (DetachVolume)

A `detachVolume` cara dikirim ke AWS akun Anda saat volume dilepaskan dari instans Amazon EC2.

Data peristiwa

Berikut ini adalah contoh `detachVolume` acara yang sukses.

```
{
  "version":"0",
```

```
  "id":"2ec37298-1234-e436-70fc-c96b1example",
  "detail-type":"AWS API Call via CloudTrail",
  "source":"aws.ec2",
  "account":"123456789012",
  "time":"2024-03-18T16:35:52Z",
  "region":"us-east-1",
  "resources":[],
  "detail":
  {
    "eventVersion":"1.09",
    "userIdentity":
    {
      "type":"IAMUser",
      "principalId":"AIDAJT12345SQ2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/administrator",
      "accountId":"123456789012",
      "accessKeyId":"AKIAJ67890A6EXAMPLE",
      "userName":"administrator"
    },
    "eventTime":"2024-03-18T16:35:52Z",
    "eventSource":"ec2.amazonaws.com",
    "eventName":"DetachVolume",
    "awsRegion":"us-east-1",
    "sourceIPAddress":"12.12.123.12",
    "userAgent":"aws-cli/2.7.12 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
ec2.detach-volume",
    "requestParameters":
    {
      "volumeId":"vol-072577c46bexample",
      "force":false
    },
    "responseElements":
    {
      "requestId":"1234513a-6292-49ea-83f8-85e95example",
      "volumeId":"vol-072577c46bexample",
      "instanceId":"i-0217f7eb3dexample",
      "device":"/dev/sdb",
      "status":"detaching",
      "attachTime":1710776815000
    },
    "requestID":"1234513a-6292-49ea-83f8-85e95example",
    "eventID":"1234551d-a15a-43eb-9e69-c983aexample",
    "readOnly":false,
    "eventType":"AwsApiCall",
```

```
    "managementEvent":true,
    "recipientAccountId":"123456789012",
    "eventCategory":"Management",
    "tlsDetails":
    {
      "tlsVersion":"TLSv1.3",
      "cipherSuite":"TLS_AES_128_GCM_SHA256",
      "clientProvidedHostHeader":"ec2.us-east-1.amazonaws.com"
    }
  }
}
```

Acara inisialisasi volume EBS

Saat Anda membuat volume Amazon EBS dari snapshot dan menggunakan tingkat inisialisasi volume default atau Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume, `initializeVolume` acara dikirim ke AWS akun Anda dalam waktu lima menit setelah inisialisasi volume selesai. Untuk informasi selengkapnya, lihat Gunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume.

Acara tidak dikirim untuk volume yang dibuat menggunakan Fast Snapshot Restore.

> **Important**
>
> `completionTime`mencerminkan kapan peristiwa itu dihasilkan. Karena acara dihasilkan dalam waktu 5 menit setelah inisialisasi selesai, `completionTime` bisa sampai lima menit setelah inisialisasi selesai.

Berikut adalah contoh kasusnya.

```
{
   "version": "0",
   "id": "01234567-0123-0123-0123-012345678901",
   "detail-type": "EBS Volume Notification",
   "source": "aws.ec2",
   "account": "012345678901",
   "time": "yyyy-mm-ddThh:mm:ssZ",
   "region": "us-east-1",
   "resources": [
      "arn:aws:ec2:us-east-1:012345678901:volume/vol-01234567890abcdef"
```

```
    ],
    "detail": {
        "event": " initializeVolume",
        "result": "succeeded",
        "completionTime": "yyyy-mm-ddThh:mm:ssZ",
        "request-id": "01234567-0123-0123-0123-0123456789ab"
    }
}
```

Peristiwa modifikasi volume EBS

Amazon EBS mengirimkan `modifyVolume` peristiwa EventBridge ketika volume dimodifikasi. Namun, itu tidak disimpan, dicatat, atau diarsipkan.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Volume Notification",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:012345678901:volume/vol-03a55cf56513fa1b6"
    ],
    "detail": {
        "result": "optimizing",
        "cause": "",
        "event": "modifyVolume",
        "request-id": "01234567-0123-0123-0123-0123456789ab"
    }
}
```

Peristiwa snapshot EBS

Amazon EBS mengirimkan peristiwa ke EventBridge saat peristiwa volume berikut terjadi.

Peristiwa

- Membuat snapshot (membuatSnapshot)
- Membuat snapshot (membuatSnapshots)
- Salin snapshot (copySnapshot)

- Bagikan snapshot (shareSnapshot)

Membuat snapshot (membuatSnapshot)

A `createSnapshot` cara dikirim ke AWS akun Anda ketika tindakan untuk membuat snapshot selesai. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Peristiwa ini dapat membawa hasil `succeeded` atau `failed`.

Data peristiwa

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS untuk peristiwa `createSnapshot` yang berhasil. Di `detail` bagian, `source` bidang berisi ARN volume sumber. Kolom `startTime` dan `endTime` mengindikasikan kapan pembuatan snapshot dimulai dan diselesaikan.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
     "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "createSnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": "arn:aws:ec2::us-west-2:volume/vol-01234567",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ"  }
}
```

Membuat snapshot (membuatSnapshots)

A `createSnapshots` cara dikirim ke AWS akun Anda ketika tindakan untuk membuat snapshot multi-volume selesai. Peristiwa ini dapat membawa hasil `succeeded` atau `failed`.

Data peristiwa

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS untuk peristiwa `createSnapshots` yang berhasil. Di bagian `detail`, bidang `source` berisi ARN volume sumber dari set snapshot multivolume. Bidang `startTime` dan `endTime` mengindikasikan kapan pembuatan snapshot dimulai dan diselesaikan.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Multi-Volume Snapshots Completion Status",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
    "arn:aws:ec2::us-east-1:snapshot/snap-012345678"
  ],
  "detail": {
    "event": "createSnapshots",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "snapshots": [
      {
        "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
        "source": "arn:aws:ec2::us-east-1:volume/vol-01234567",
        "status": "completed"
      },
      {
        "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-012345678",
        "source": "arn:aws:ec2::us-east-1:volume/vol-012345678",
        "status": "completed"
      }
    ]
  }
}
```

Daftar di bawah ini adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `createSnapshots` yang gagal. Penyebab kegagalan adalah satu atau lebih snapshot untuk set

snapshot multi-volume gagal diselesaikan. Nilai-nilai `snapshot_id` adalah ARNs snapshot yang gagal. `startTime` dan `endTime` mewakili kapan tindakan create-snapshots dimulai dan berakhir.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Multi-Volume Snapshots Completion Status",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
    "arn:aws:ec2::us-east-1:snapshot/snap-012345678"
  ],
"detail": {
    "event": "createSnapshots",
    "result": "failed",
    "cause": "Snapshot snap-01234567 is in status error",
   "request-id": "",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "snapshots": [
      {
        "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
        "source": "arn:aws:ec2::us-east-1:volume/vol-01234567",
        "status": "error"
      },
      {
        "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-012345678",
        "source": "arn:aws:ec2::us-east-1:volume/vol-012345678",
        "status": "error"
      }
    ]
  }
}
```

Salin snapshot (copySnapshot)

A `copySnapshot` cara dikirim ke AWS akun Anda ketika tindakan untuk menyalin snapshot selesai. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Peristiwa ini dapat membawa hasil `succeeded` atau `failed`.

Di `detail` bagian tersebut, `source` adalah ARN dari snapshot sumber, dan `snapshot_id` merupakan ARN dari salinan snapshot. `startTime`dan `endTime` menunjukkan kapan operasi penyalinan dimulai dan berakhir. `incremental`menunjukkan apakah salinan snapshot adalah snapshot inkremental (`true`), atau snapshot penuh (`false`). `transferType`menunjukkan apakah operasi penyalinan snapshot adalah operasi penyalinan standar atau operasi penyalinan berbasis waktu. Untuk informasi selengkapnya, lihat Time-based salinan untuk snapshot dan AMI Amazon EBS EBS-backed.

Jika Anda menyalin snapshot di seluruh Wilayah, peristiwa tersebut dipancarkan di Wilayah tujuan.

Skenario 1: Operasi penyalinan snapshot standar selesai

Berikut ini adalah contoh peristiwa yang dikirim ke akun Anda ketika operasi penyalinan snapshot standar berhasil diselesaikan. Perhatikan bahwa `transferType` itu`standard`.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "copySnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": "arn:aws:ec2::eu-west-1:snapshot/snap-76543210",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "incremental": "true",
    "transferType": "standard"
  }
}
```

Skenario 2: operasi Time-based penyalinan snapshot selesai dalam durasi penyelesaian

Berikut ini adalah contoh peristiwa yang dikirim ke akun Anda ketika operasi penyalinan snapshot berbasis waktu selesai dalam durasi penyelesaiannya. Perhatikan `transferType` itu `time-based` untuk menunjukkan bahwa itu adalah operasi penyalinan snapshot berbasis waktu. `completionDurationStartTime`menunjukkan kapan durasi penyelesaian dimulai.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "copySnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": "arn:aws:ec2::eu-west-1:snapshot/snap-76543210",
    "incremental": "true",
    "completionDurationStartTime":"2024-11-16T06:27:33.816Z",
    "transferType": "time-based"
  }
}
```

Skenario 3: operasi Time-based penyalinan snapshot selesai tetapi melewatkan durasi penyelesaian yang diminta

Ketika operasi penyalinan snapshot berbasis waktu selesai, tetapi gagal memenuhi durasi penyelesaian yang diminta, CloudWatch mengirimkan dua peristiwa ke akun Anda. Berikut ini adalah contoh dari peristiwa tersebut.

- Acara pertama dikirim ke akun Anda segera setelah durasi penyelesaian terlewatkan, bahkan jika operasi penyalinan masih berlangsung. Untuk acara ini, `detail-type` ada`EBS Copy Snapshot`

`Missed Completion Duration`, dan `missedCompletionDurationCause` memberikan alasannya.

```
{
 "version":"0",
 "id":"fd90eb95-0938-e02c-cf55-b81363b8ac12",
 "detail-type":"EBS Copy Snapshot Missed Completion Duration",
 "source":"aws.ec2",
 "account":"123456789012",
 "time":"2024-11-19T18:17:08Z",
 "region":"us-east-1",
 "resources": ["arn:aws:ec2:us-east-1:123456789012:snapshot/snap-01234567890abcedf"],
 "detail":{
   "event":"copySnapshot",
   "missedCompletionDurationCause":"Snapshot copy was not able to meet the specified
 completion duration because your snapshot copy operation throughput quota was
 exceeded.",
   "snapshot_id":"arn:aws:ec2:us-
east-1:123456789012:snapshot/snap-01234567890abcedf",
   "source":"arn:aws:ec2:us-east-1:123456789012:snapshot/snap-00987654321fedcba",
   "startTime": "Sun Nov 24 22:32:55 UTC 2024",
   "transferType": "time-based"
   }
}
```

- Acara kedua dikirim ke akun Anda hanya setelah snapshot selesai. Acara termasuk`missedCompletionDurationCause`, yang memberikan alasannya.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "copySnapshot",
    "result": "succeeded",
    "cause": "",
```

```
    "request-id": "",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": "arn:aws:ec2::eu-west-1:snapshot/snap-76543210",
    "incremental": "true",
    "completionDurationStartTime":"2024-11-16T06:27:33.816Z",
    "missedCompletionDurationCause":"Snapshot copy was not able to meet the specified
 completion duration because your snapshot copy operation throughput quota was
 exceeded.",
    "transferType": "time-based"
  }
}
```

Skenario 4: Operasi salinan snapshot gagal

Berikut ini adalah contoh peristiwa yang dikirim ke akun Anda ketika operasi penyalinan snapshot gagal. Perhatikan `result` itu `failed` untuk menunjukkan bahwa operasi gagal.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "copySnapshot",
    "result": "failed",
    "cause": "Source snapshot ID is not valid",
    "request-id": "",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": "arn:aws:ec2::eu-west-1:snapshot/snap-76543210",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ"
  }
}
```

Bagikan snapshot (shareSnapshot)

A `shareSnapshot` cara dikirim ke AWS akun Anda ketika akun lain membagikan snapshot dengannya. Namun, itu tidak disimpan, dicatat, atau diarsipkan. Hasilnya selalu `succeeded`.

Data peristiwa

Berikut adalah contoh objek JSON yang dipancarkan oleh EBS setelah peristiwa `shareSnapshot` yang gagal. Di `detail` bagian tersebut, nilai `source` adalah nomor AWS akun pengguna yang berbagi snapshot dengan Anda. `startTime`dan `endTime` mewakili kapan tindakan share-snapshot dimulai dan berakhir. Peristiwa `shareSnapshot` ini hanya akan dilakukan saat snapshot privat dibagikan dengan pengguna lain. Berbagi snapshot publik tidak memicu peristiwa.

```
{
  "version": "0",
  "id": "01234567-01234-0123-0123-012345678901",
  "detail-type": "EBS Snapshot Notification",
  "source": "aws.ec2",
  "account": "012345678901",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ec2::us-west-2:snapshot/snap-01234567"
  ],
  "detail": {
    "event": "shareSnapshot",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "snapshot_id": "arn:aws:ec2::us-west-2:snapshot/snap-01234567",
    "source": 012345678901,
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ"
  }
}
```

Peristiwa Arsip Snapshots EBS

Amazon EBS memancarkan peristiwa yang terkait dengan tindakan pengarsipan snapshot. Untuk informasi selengkapnya, lihat Memantau pengarsipan snapshot Amazon EBS menggunakan Acara CloudWatch.

Peristiwa pemulihan snapshot cepat EBS

Amazon EBS mengirimkan peristiwa ke EventBridge saat status pemulihan snapshot cepat untuk snapshot berubah. Peristiwa dipancarkan atas dasar upaya terbaik.

Berikut adalah contoh data untuk peristiwa ini.

```
{
   "version": "0",
   "id": "01234567-0123-0123-0123-012345678901",
   "detail-type": "EBS Fast Snapshot Restore State-change Notification",
   "source": "aws.ec2",
   "account": "123456789012",
   "time": "yyyy-mm-ddThh:mm:ssZ",
   "region": "us-east-1",
   "resources": [
      "arn:aws:ec2:us-east-1::snapshot/snap-03a55cf56513fa1b6"
   ],
   "detail": {
      "snapshot-id": "snap-1234567890abcdef0",
      "state": "optimizing",
      "zone": "us-east-1a",
      "message": "Client.UserInitiated - Lifecycle state transition",
   }
}
```

Kemungkinan nilai untuk `state` adalah `enabling`, `optimizing`, `enabled`, `disabling`, dan `disabled`.

Kemungkinan nilai untuk `message` adalah sebagai berikut:

`Client.InvalidSnapshot.InvalidState - The requested snapshot transitioned to an invalid state (Error)`

Permintaan untuk mengaktifkan pemulihan snapshot cepat gagal dan status bertransisi ke `disabling` atau `disabled`. Pemulihan snapshot cepat tidak diaktifkan untuk snapshot ini.

`Client.UserInitiated`

Status berhasil dialihkan ke `enabling` atau `disabling`.

`Client.UserInitiated - Lifecycle state transition`

Status berhasil dialihkan ke `optimizing`, `enabled`, atau `disabled`.

`Server.InsufficientCapacity - There was insufficient capacity available to satisfy the request`

Permintaan untuk mengaktifkan pemulihan snapshot cepat gagal karena kapasitas tidak mencukupi, dan status bertransisi ke `disabling` atau `disabled`. Tunggu dan kemudian coba lagi.

`Server.InternalError - An internal error caused the operation to fail`

Permintaan untuk mengaktifkan pemulihan snapshot cepat gagal karena kesalahan internal, dan status bertransisi ke `disabling` atau `disabled`. Tunggu dan kemudian coba lagi.

`Client.InvalidSnapshot.InvalidState - The requested snapshot was deleted or access permissions were revoked`

Status pemulihan snapshot cepat untuk snapshot telah ditransisikan ke `disabling` atau `disabled` karena snapshot dihapus atau tidak dibagikan oleh pemilik snapshot. Pemulihan snapshot cepat tidak dapat diaktifkan untuk snapshot yang telah dihapus atau tidak lagi dibagikan kepada Anda.

Penggunaan AWS Lambda untuk menangani EventBridge peristiwa

Anda dapat menggunakan Amazon EBS dan Amazon EventBridge untuk mengotomatiskan alur kerja pencadangan data Anda. Ini mengharuskan Anda membuat kebijakan IAM, AWS Lambda fungsi untuk menangani peristiwa, dan EventBridge aturan yang cocok dengan peristiwa masuk dan merutekan mereka ke fungsi Lambda.

Prosedur berikut menggunakan peristiwa `createSnapshot` untuk menyalin snapshot yang sudah selesai secara otomatis ke Wilayah lain untuk pemulihan bencana.

Untuk menyalin snapshot yang sudah selesai ke Wilayah lain

1. Buat kebijakan IAM, seperti yang ditunjukkan dalam contoh berikut, untuk memberikan izin untuk menggunakan `CopySnapshot` tindakan dan menulis ke EventBridge log. Tetapkan kebijakan kepada pengguna yang akan menangani EventBridge acara tersebut.

   JSON

```
{
  "Version":"2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CopySnapshot"
      ],
      "Resource": "*"
    }
  ]
}
```

2. Tentukan fungsi di Lambda yang akan tersedia dari EventBridge konsol. Contoh fungsi Lambda di bawah ini, ditulis dalam Node.js, dipanggil oleh EventBridge ketika peristiwa pencoco `createSnapshot` kan dipancarkan oleh Amazon EBS (menandakan bahwa snapshot telah selesai). Saat diinvokasi, fungsi menyalin snapshot dari `us-east-2` untuk `us-east-1`.

```
// Sample Lambda function to copy an EBS snapshot to a different Region

var AWS = require('aws-sdk');
var ec2 = new AWS.EC2();

// define variables
var destinationRegion = 'us-east-1';
var sourceRegion = 'us-east-2';
console.log ('Loading function');

//main function
exports.handler = (event, context, callback) => {

    // Get the EBS snapshot ID from the event details
    var snapshotArn = event.detail.snapshot_id.split('/');
    const snapshotId = snapshotArn[1];
    const description = `Snapshot copy from ${snapshotId} in ${sourceRegion}.`;
    console.log ("snapshotId:", snapshotId);
```

```
    // Load EC2 class and update the configuration to use destination Region to
 initiate the snapshot.
    AWS.config.update({region: destinationRegion});
    var ec2 = new AWS.EC2();

    // Prepare variables for ec2.modifySnapshotAttribute call
    const copySnapshotParams = {
        Description: description,
        DestinationRegion: destinationRegion,
        SourceRegion: sourceRegion,
        SourceSnapshotId: snapshotId
    };

    // Execute the copy snapshot and log any errors
    ec2.copySnapshot(copySnapshotParams, (err, data) => {
        if (err) {
            const errorMessage = `Error copying snapshot ${snapshotId} to Region
 ${destinationRegion}.`;
            console.log(errorMessage);
            console.log(err);
            callback(errorMessage);
        } else {
            const successMessage = `Successfully started copy of snapshot
 ${snapshotId} to Region ${destinationRegion}.`;
            console.log(successMessage);
            console.log(data);
            callback(null, successMessage);
        }
    });
};
```

Untuk memastikan bahwa fungsi Lambda Anda tersedia dari EventBridge konsol, buat di Wilayah tempat EventBridge kejadian akan terjadi. Lihat informasi selengkapnya di Panduan Developer AWS Lambda.

3. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
4. Di panel navigasi, pilih buat Aturan, lalu pilih Buat aturan.
5. Untuk Langkah 1: Tentukan detail aturan, lakukan hal berikut ini:
   a. Masukkan nilai untuk Nama dan Deskripsi.
   b. Untuk Bus peristiwa, tetap default.

c. Pastikan Aktifkan aturan pada bus acara yang dipilih diaktifkan.

d. Untuk Tipe peristiwa, pilih Aturan dengan pola peristiwa.

e. Pilih Berikutnya.

6. Untuk Langkah 2: Bangun pola peristiwa, lakukan hal berikut ini:

a. Untuk Sumber acara, pilih AWS acara atau acara EventBridge mitra.

b. Di bagian Pola peristiwa, untuk Sumber peristiwa, pastikan AWS layanan dipilih, dan untuk AWS layanan, pilih EC2.

c. Untuk Tipe peristiwa, pilih Notifikasi Snapshot EBS, pilih Peristiwa spesifik, lalu pilih createSnapshot.

d. Pilih Hasil spesifik dan kemudian pilih berhasil.

e. Pilih Berikutnya.

7. Untuk Langkah 3: Pilih target, lakukan hal berikut ini:

a. Untuk Tipe target, pilih Layanan AWS .

b. Untuk Pilih target, pilih fungsi Lambda, dan untuk Fungsi pilih fungsi yang Anda buat sebelumnya.

c. Pilih Selanjutnya

8. Untuk Langkah 4: Konfigurasikan tanda, tentukan tanda untuk aturan jika diperlukan, lalu pilih Berikutnya.

9. Untuk Langkah 5: Tinjau dan buat, tinjau aturan lalu pilih Buat aturan.

Aturan Anda seharusnya kini muncul di tab Aturan. Pada contoh yang ditunjukkan, peristiwa yang Anda konfigurasikan harusoleh EBS pada saat Anda menyalin snapshot berikutnya.

Statistik kinerja terperinci Amazon EBS

Perangkat blok Amazon EBS NVMe menjual statistik I/O kinerja resolusi tinggi real-time untuk volume Amazon EBS yang terpasang pada instans Amazon EC2. Nitro-based Statistik ini disajikan sebagai penghitung agregat yang dipertahankan selama lampiran volume ke instans. Statistik memberikan rincian tentang jumlah operasi kumulatif, byte yang dikirim dan diterima, dan waktu yang dihabiskan untuk I/O operasi baca dan tulis. Selain itu, statistik termasuk histogram untuk operasi baca dan tulis I/O . Mereka juga melaporkan total waktu aplikasi Anda telah melampaui batas IOPS atau throughput yang disediakan volume EBS atau instans terlampir.

Anda dapat mengumpulkan statistik ini dengan perincian hingga interval 1 detik. Jika permintaan dibuat lebih sering dari interval 1 detik, driver NVMe mungkin mengantri permintaan, bersama dengan perintah admin lainnya, untuk diproses di lain waktu.

Pertimbangan-pertimbangan

- Statistik didukung untuk semua jenis volume Amazon EBS.
- Statistik hanya didukung untuk volume yang dilampirkan ke instans yang dibangun di Sistem AWS Nitro.
- Statistik tersedia untuk volume yang Multi-Attach diaktifkan. Saat melihat statistik untuk volume yang Multi-Attach diaktifkan, statistik khusus untuk lampiran instans tersebut, dan hanya mencerminkan penggunaan instance tersebut.
- Statistik tersedia tanpa biaya tambahan.

Statistik

Perangkat blok Amazon EBS NVMe menjual statistik berikut:

Nama statistik	Nama lengkap	Tipe	Deskripsi
`total_rea d_ops`	Total operasi baca	Penghitun g	Jumlah total operasi baca yang selesai.
`total_wr te_ops`	Total operasi penulisan	Penghitun g	Jumlah total operasi penulisan yang diselesai kan.
`total_rea d_bytes`	Jumlah byte yang dibaca	Penghitun g	Jumlah total byte baca yang ditransfer.
`total_wr te_bytes`	Jumlah byte tulis	Penghitun g	Jumlah total byte tulis yang ditransfer.
`total_rea d_time`	Total waktu baca	Penghitun g	Total waktu yang dihabiskan, dalam mikrodetik, oleh semua operasi baca yang selesai.

Nama statistik	Nama lengkap	Tipe	Deskripsi
`total_wr: te_time`	Total waktu penulisan	Penghitung	Total waktu yang dihabiskan, dalam mikrodetik, oleh semua operasi penulisan yang diselesaikan.
`ebs_volu e_perfor ance_exc eded_iop`	Total permintaan waktu melebihi volume yang disediakan IOPS	Penghitung	Total waktu, dalam mikrodetik, permintaan IOPS melebihi kinerja IOPS yang disediakan volume.
`ebs_volu e_perfor ance_exc eded_tp`	Total permintaan waktu melebihi volume throughput yang disediakan	Penghitung	Total waktu, dalam mikrodetik, permintaan throughput melebihi kinerja throughput yang disediakan volume.
`ec2_inst nce_ebs_ erforman e_exceed d_iops`	Total permintaan waktu melebihi kinerja IOPS instans EC2	Penghitung	Total waktu, dalam mikrodetik, volume EBS melebihi kinerja IOPS maksimum instans Amazon EC2 yang terpasang.
`ec2_inst nce_ebs_ erforman e_exceed d_tp`	Total permintaan waktu melebihi kinerja throughput instans EC2	Penghitung	Total waktu, dalam mikrodetik, volume EBS melebihi kinerja throughput maksimum instans Amazon EC2 yang terpasang.
`volume_q eue_leng h`	Panjang antrian volume	Titik waktu	Jumlah operasi baca dan tulis yang menunggu untuk diselesaikan.
`read_io_ atency_h stogram`	Baca I/O histogram	Histogram *	Jumlah operasi baca yang diselesaikan dalam setiap bin latensi, dalam mikrodetik.

Nama statistik	Nama lengkap	Tipe	Deskripsi
`write_io_latency_histogram`	Tulis I/O histogram	Histogram *	Jumlah operasi tulis yang diselesaikan dalam setiap bin latensi, dalam mikrodetik.

> **Note**
>
> * Statistik histogram hanya mewakili I/O operasi yang telah berhasil diselesaikan. I/O Operasi yang terhenti atau terganggu tidak termasuk, tetapi akan terlihat dalam `volume_queue_length` statistik, yang disajikan sebagai statistik point-in-time.

Mengakses statistik

Statistik harus diakses langsung dari instance yang dilampirkan volume Amazon EBS. Anda dapat mengakses statistik menggunakan salah satu metode berikut.

Instans Linux

Amazon CloudWatch

Anda dapat mengonfigurasi CloudWatch agen Amazon untuk mengumpulkan statistik dari instans Anda dan membuatnya tersedia sebagai metrik khusus di CloudWatch. Anda kemudian dapat menggunakan metrik CloudWatch untuk menganalisis I/O pola, melacak tren kinerja, membuat dasbor khusus, dan mengatur alarm otomatis berdasarkan ambang kinerja.

Untuk informasi selengkapnya tentang mengonfigurasi CloudWatch agen, lihat berikut ini:

- Buat file konfigurasi CloudWatch agen
- Kumpulkan metrik driver Amazon EBS NVMe

Dengan versi add-on Amazon CloudWatch Observability EKS `4.1.0` dan yang lebih baru, statistik dikumpulkan secara otomatis saat metrik driver Amazon EBS CSI diaktifkan. Untuk informasi selengkapnya, lihat met rik driver Amazon EBS NVMe.

ebsnvme script

`ebsnvme`Skrip dapat ditemukan di repo Github amazon-ec2-utils.

Untuk mengakses statistik

1. Hubungkan ke instance tempat volume terpasang.
2. Unduh `ebsnvme` skrip dari repo `amazon-ec2-utils` Github.

```
wget https://raw.githubusercontent.com/amazonlinux/amazon-ec2-utils/refs/heads/main/ebsnvme
```

3. Ubah izin untuk skrip untuk membuatnya dapat dieksekusi.

```
sudo chmod +x ./ebsnvme
```

4. Jalan `ebsnvme` kan skrip dan tentukan nama perangkat untuk volume.

```
sudo ./ebsnvme stats /dev/nvme0n1
```

nvme-cli tool

Untuk mengakses statistik

1. Hubungkan ke instance tempat volume terpasang.
2. AMI Amazon Linux yang dirilis setelah 12 November 2024 menyertakan versi terbaru dari `nvme-cli` alat ini. Jika Anda menggunakan AMI Amazon Linux yang lebih lama, perbarui `nvme-cli` alat ini.

```
sudo yum install nvme-cli
```

3. Jalankan perintah berikut dan tentukan nama perangkat untuk volume.

```
nvme amzn stats /dev/nvme0n1
```

Prometheus

Anda dapat memantau statistik dengan Prometheus, aplikasi pemantauan sumber terbuka, dan Amazon Managed Service for Prometheus. Hal ini memudahkan pemantauan volume

Amazon EBS di seluruh wadah dan lingkungan Kubernetes dalam skala besar. Dengan driver Amazon EBS CSI versi v1.37.0 dan yang lebih baru, statistik kinerja terperinci diekspos sebagai Prometheus-compatible `/metrics` titik akhir untuk mengekspor ke Prometheus.

Untuk informasi selengkapnya, lihat Meng ambil metrik ke ruang kerja Layanan Terkelola Amazon untuk Prometheus di Panduan Pengguna Layanan Terkelola Amazon untuk Prometheus.

Instans Windows

nvme_amzn.exe tool

Untuk mengakses statistik

1. Hubungkan ke instance tempat volume terpasang.
2. Pastikan Anda menggunakan versi `1.7.0` driver AwsnVMe atau yang lebih baru. Untuk informasi selengkapnya tentang memperbarui driver AWSNVMe, lihat AWS driver NVMe.
3. Dapatkan nomor disk untuk volume EBS. Untuk informasi selengkapnya, lihat Mem etakan volume Amazon EBS ke nama perangkat NVMe
4. Jalankan perintah berikut sebagai Administrator dan tentukan nomor disk untuk volume.

```
.\nvme_amzn.exe stats disk_number
```

Amazon GuardDuty untuk Amazon EBS

Amazon GuardDuty adalah layanan deteksi ancaman yang membantu melindungi akun, wadah, beban kerja, dan data di AWS lingkungan Anda. Menggunakan model machine learning (ML), serta kemampuan deteksi anomali dan ancaman, GuardDuty terus memantau berbagai sumber log dan aktivitas runtime untuk mengidentifikasi dan memprioritaskan potensi risiko keamanan dan aktivitas berbahaya di lingkungan Anda.

Fitur Perlindungan Malware dalam GuardDuty memindai volume Amazon EBS yang terkait dengan instans Amazon EC2 dan beban kerja container untuk mendeteksi potensi ancaman. GuardDuty menawarkan dua cara untuk melakukan ini:

- Aktifkan Perlindungan Malware — Saat GuardDuty menghasilkan temuan yang menunjukkan potensi keberadaan malware di instans Amazon EC2 atau beban kerja kontainer, ia akan secara otomatis memulai pemindaian malware pada sumber daya yang berpotensi disusupi.
- Gunakan pemindaian malware sesuai permintaan tanpa mengaktifkan Perlindungan Malware — Berikan Nama Sumber Daya Amazon (ARN) instans Amazon EC2 Anda untuk memulai pemindaian sesuai permintaan.

Untuk informasi selengkapnya, lihat Panduan GuardDuty Pengguna Amazon.

Memahami kode untuk Amazon EBS dalam laporan penagihan dan penggunaan

Saat Anda menggunakan Amazon EBS, kami menyertakan kode terkait dalam laporan AWS penagihan dan penggunaan Anda. Meninjau kode ini membantu Anda memahami biaya dan pola penggunaan untuk Amazon EBS. Melacak dan mengelola pengeluaran Anda sangat penting untuk mengoptimalkan biaya Anda.

Tabel berikut menjelaskan kode untuk Amazon EBS yang muncul di laporan penagihan dan penggunaan Anda. Untuk daftar kode Wilayah yang digunakan dalam laporan penagihan dan penggunaan, lihat Kode penagihan AWS wilayah.

Kode penagihan untuk:

- Snapshot
- Penyimpanan volume
- Kinerja yang disediakan
- API langsung EBS
- Pemulihan snapshot cepat (FSR)

Sumber daya terkait

- Harga Amazon EBS
- the section called “Harga”

Snapshot

Kode	Deskripsi	Unit	Granularitas
region-EBS:SnapshotUsage	Snapshot tingkat standar (salinan tambahan).	GB-Month	Per Jam
region-EBS:SnapshotArchiveStorage	Arsipkan snapshot tingkat (salinan lengkap).	GB-Month	Per Jam
region-EBS:SnapshotArchiveEarlyDelete	Menghapus snapshot yang diarsipkan sebelum periode retensi minimum berakhir.	GB	Per penghapusan
region-EBS:SnapshotArchiveRetrieval	Mengambil snapshot yang diarsipkan.	GB	Per GB

Penyimpanan volume

Kode	Deskripsi	Unit	Granularitas
region-EBS:VolumeUsage.gp3	Penyimpanan untuk volume General Purpose SSD (gp3).	GB-Month	Per Jam
region-EBS:Volu	Penyimpanan untuk volume General Purpose SSD (g2).	GB-Month	Per Jam

Kode	Deskripsi	Unit	Granularitas
meUsage.gp2			
region-EBS:VolumeUsage.io2	Penyimpanan untuk volume IOPS SSD (io2) yang Disediakan.	GB-Month	Per Jam
region-EBS:VolumeUsage.piops	Penyimpanan untuk volume IOPS SSD (io1) yang Disediakan.	GB-Month	Per Jam
region-EBS:VolumeUsage.st1	Penyimpanan untuk volume HDD yang Dioptimalkan Throughput.	GB-Month	Per Jam
region-EBS:VolumeUsage.sc1	Penyimpanan untuk volume HDD Dingin.	GB-Month	Per Jam
region-EBS:VolumeUsage	Penyimpanan untuk volume Magnetik.	GB-Month	Per Jam

Kinerja yang disediakan

Kode	Deskripsi	Unit	Granularitas
region-EBS:Volu	Throughput yang disediakan untuk volume gp3 lebih dari 125. MB/s	MB/s-Month	Per Jam

Kode	Deskripsi	Unit	Granularitas
meP-Throughput.gp3			
region-EBS:VolumeP-IOPS.gp3	IOPS yang disediakan untuk volume gp3 lebih dari 3.000.	IOPS-Month	Per Jam
region-EBS:VolumeP-IOPS.io2	IOPS yang disediakan untuk volume io2 hingga 32.000.	IOPS-Month	Per Jam
region-EBS:VolumeP-IOPS.io2.tier2	IOPS yang disediakan untuk volume io2 dari 32.001 hingga 64.000.	IOPS-Month	Per Jam
region-EBS:VolumeP-IOPS.io2.tier3	IOPS yang disediakan untuk volume io2 lebih besar dari 64.000.	IOPS-Month	Per Jam
region-EBS:VolumeP-IOPS.piops	IOPS yang disediakan untuk volume io1.	IOPS-Month	Per Jam
region-EBS:VolumeIOUsage	IOPS Warisan.	IOPS-Month	Per Jam

API langsung EBS

Kode	Deskripsi	Unit	Granularitas
region-EBS:directAPI.snapshot.List	Panggilan ke tindakan ListChangedBlocks dan ListSnapshotBlocks API.	Per 1.000 permintaan	Per permintaan
region-EBS:directAPI.snapshot.Get	Blok yang dikembalikan oleh aksi GetSnapshotBlock API.	Per 1.000 blok dikembalikan	Per permintaan
region-EBS:directAPI.snapshot.Put	Blok yang ditulis oleh aksi PutSnapshotBlock API.	Per 1.000 blok ditulis	Per permintaan

Pemulihan snapshot cepat (FSR)

Kode	Deskripsi	Unit	Granularitas
region-EBS:FastSnapshotRestore	Layanan Data Unit-Hours (DSU-Hours) untuk snapshot diaktifkan untuk pemulihan snapshot cepat (per AZ).	DSU-Hours	Per menit (minimal satu jam)

Buat inventaris volume EBS Anda

Amazon EBS menyediakan sumber daya penyimpanan blok berkinerja tinggi yang dapat diskalakan yang dapat digunakan dengan instans EC2. Anda dapat melampirkan volume EBS ke instans EC2. Anda juga dapat melepaskan volume EBS dari instans EC2.

Berikut ini adalah karakteristik utama penyimpanan EBS. Administrator sistem dapat memperoleh informasi tentang karakteristik ini untuk sumber daya EBS Anda dan menggunakannya untuk mengonfigurasi penyimpanan yang setara secara fungsional untuk server yang Anda jalankan di tempat atau server dari Penyedia Cloud lain. Administrator sistem juga dapat mengambil data yang disimpan di sumber daya EBS Anda dan kemudian menyimpan data di tempat atau di penyimpanan dari Penyedia Cloud lain.

Karakteristik	Deskripsi
Volume	Volume EBS adalah perangkat penyimpanan tingkat blok yang tahan lama yang dapat Anda lampirkan ke instance, format, dan pemasanga n.
I/O kinerja	Amazon EBS menyediakan beberapa jenis volume, yang berbeda dalam hal karakteristik kinerja dan harga.
Isi volume data EBS Anda	Satu-satunya cara untuk mengakses data pada volume secara langsung adalah dari instance.
Cuplikan foto	Snapshot EBS adalah salinan point-in-time dari data pada volume EBS. Snapshot disimpan di Amazon S3, dalam bucket yang tidak dapat Anda akses secara langsung.
Isi snapshot EBS Anda	Anda dapat membaca data dari snapshot Anda, atau menggunakannya untuk membuat volume.

volume

Saat Anda membuat volume EBS, Anda memilih Zona Ketersediaan untuk volume tersebut. Anda harus melampirkan volume EBS ke instans EC2 di Zona Ketersediaan yang sama. Anda dapat membuat volume saat meluncurkan instance EC2, atau membuatnya terlebih dahulu lalu melampirkannya ke instans. Jika volume dilampirkan ke instance, statusnya adalah`in-use`. Jika volume tidak dilampirkan ke instance, statusnya adalah`available`.

Mengapa ini penting

Volume root EBS berisi sistem operasi untuk instans EC2 Anda. Volume data EBS dapat berisi data penting bisnis. Anda dapat membuat daftar volume EBS di semua Wilayah dan apakah volume tersebut dilampirkan ke instans EC2.

Untuk mendapatkan ringkasan volume EBS Anda di semua Wilayah

Anda dapat menggunakan Amazon EC2 Global View untuk membuat daftar volume EBS di semua Wilayah.

1. Buka konsol Amazon EC2 Global View di https://console.aws.amazon.com/ec2globalview/home.
2. Pada tab Penjelajah wilayah, di bawah Ring kasan, periksa jumlah sumber daya untuk Volume, yang mencakup jumlah volume dan jumlah Wilayah. Pilih teks yang digarisbawahi untuk melihat bagaimana jumlah volume tersebar di seluruh Wilayah.
3. Pada tab Pencarian global, pilih filter klien Jenis sumber daya = Volume. Anda dapat memfilter hasil lebih lanjut dengan menentukan Region atau tag.

Untuk menggambarkan volume EBS yang digunakan

Gunakan perintah describe-volumes. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki volume EBS. `--filters`Parameter mencakup hasil ke volume yang ada`in-use`. `--query`Parameter hanya menampilkan bidang yang ditentukan dalam output. Anda dapat menyertakan bidang tambahan sesuai kebutuhan.

```
aws ec2 describe-volumes \
    --filters Name=status,Values=in-use \
    --query "Volumes[].[VolumeId,Size,Attachments[0].InstanceId]" \
    --output table
```

Berikut ini adalah output contoh. Kolom adalah ID volume, ukuran volume, ID instance terlampir, dan nama perangkat.

```
-------------------------------------------------------------------
|                          DescribeVolumes                        |
+-------------------------+-----+----------------------+---------+
|  vol-0992ee65bec96de19 |  8  |  i-00a7d9ec76a46a49f  |  /dev/xvda |
|  vol-04d631d22047db1cb |  30 |  i-0b1bf24fd4f297ab9  |  /dev/sda1 |
   ...
   ...
|  vol-0b811be5cbd1425cc | 100 |  i-0a8d998154b320257  |  /dev/xvde |
```

```
+------------------------+-----+----------------------+------------+
```

Untuk menggambarkan volume EBS yang tidak digunakan

Pada perintah sebelumnya, ubah filter untuk menemukan volume dengan status `available` sebagai berikut.

```
--filters Name=status,Values=available
```

Jika volume tidak digunakan, Anda harus menentukan apakah volume tersebut memiliki data yang Anda butuhkan. Misalnya, Anda dapat menghapus volume yang dibuat untuk tujuan pengujian atau untuk memecahkan masalah. Volume mungkin juga tersedia jika perangkat blok yang melampirkan volume ke instance di konfigurasi untuk bertahan setelah penghentian instance. Jika ini masalahnya, verifikasi apakah data pada volume masih diperlukan.

I/O kinerja

Jenis volume yang Anda tentukan untuk volume EBS menentukan kinerjanya. Bergantung pada jenis volume, Anda dapat menentukan kombinasi ukuran, IOPS, dan throughput untuk volume Anda. Untuk informasi tentang IOPS dan throughput maksimum yang disediakan untuk setiap jenis volume, lihat jenis volume Amazon EBS.

Mengapa hal ini penting

Setelah menentukan kinerja volume EBS, Anda dapat menentukan spesifikasi minimum yang diperlukan untuk penyimpanan yang setara secara fungsional.

Untuk mendapatkan informasi tentang kinerja volume Anda

Gunakan perintah describe-volume untuk menggambarkan kinerja volume yang sedang digunakan. `--query`Parameter hanya menampilkan bidang yang ditentukan dalam output. Anda dapat menyertakan bidang tambahan sesuai kebutuhan. Untuk informasi tentang EBS-optimized kinerja, lihat spesifikasi jenis instans Amazon EC2 di Panduan Jenis Instans Amazon EC2.

```
aws ec2 describe-volumes \
    --filters Name=status,Values=in-use \
    --query "Volumes[].[VolumeId,VolumeType,Iops,Throughput]" \
    --output table
```

Berikut ini adalah output contoh. Kolom adalah ID volume, tipe volume, IOPS, dan throughput (MiB/s).

```
-------------------------------------------------
|                DescribeVolumes                |
+------------------------+------+-------+-------+
|  vol-0992ee65bec96de19 |  gp3 |  3000 |  125  |
|  vol-04d631d22047db1cb |  gp2 |  100  |  None |
   ...
   ...
|  vol-0b811be5cbd1425cc |  gp3 |  3000 |  125  |
+------------------------+------+-------+-------+
```

Isi volume data EBS Anda

Amazon EBS tidak memiliki akses ke data yang Anda simpan di volume EBS Anda. Anda bertanggung jawab untuk membuat cadangan volume data Anda. Misalnya, Anda dapat membuat snapshot point-in-time, yang dikenal sebagai snapshot EBS.

Anda juga dapat terhubung ke instans EC2 dan mentransfer file dari volume EBS ke komputer Anda sendiri. Ada banyak alat yang tersedia untuk membantu Anda melakukan ini. Berikut ini adalah beberapa contohnya.

Mengapa hal ini penting

Jika Anda memiliki data penting bisnis pada volume EBS Anda, Anda dapat menyalinnya ke penyimpanan yang setara secara fungsional.

Untuk mentransfer file dari volume yang dilampirkan ke instance Linux

Hubungkan ke instance Linux Anda dan gunakan SCP. Untuk informasi selengkapnya, lihat Mentransfer file menggunakan SCP.

Untuk mentransfer file dari volume yang dilampirkan ke instance Windows

Hubungkan ke instance Windows Anda dan gunakan RDP. Untuk informasi selengkapnya, lihat Mentransfer file menggunakan RDP.

Snapshot

Anda dapat mencadangkan volume data EBS Anda dengan membuat snapshot EBS. Anda dapat membuat snapshot EBS menggunakan Amazon EBS, Amazon Data Lifecycle Manager, atau. AWS Backup

Mengapa hal ini penting

Jika Anda memiliki data penting bisnis pada volume EBS Anda, kemungkinan besar Anda membuat cadangan dalam bentuk snapshot EBS. Anda dapat menemukan snapshot terbaru dari volume Anda dan memverifikasi bahwa mereka mencerminkan data saat ini pada volume.

Untuk mendeskripsikan snapshot untuk volume tertentu

Gunakan perintah describe-snapshots. `--filters`Parameter mencakup hasil ke snapshot untuk volume yang ditentukan. `--query`Parameter hanya menampilkan bidang yang ditentukan dalam output. Anda dapat menyertakan bidang tambahan sesuai kebutuhan.

```
aws ec2 describe-snapshots \
    --filters Name=volume-id,Values=vol-00622ef8c2ac8b762 \
    --query Snapshots[*].[SnapshotId,CompletionTime,Description] --output table
```

Berikut ini adalah output contoh. Kolom adalah ID snapshot, tanggal penyelesaian, dan apakah volume dienkripsi.

```
-------------------------------------------------------------------------
|                            DescribeSnapshots                          |
+-------------------------+-----------------------------------+--------+
|  snap-0ad439c50efabb47c |  2024-07-12T03:36:27.952000+00:00  |  False |
+-------------------------+-----------------------------------+--------+
```

Isi snapshot EBS Anda

Anda tidak dapat mengunduh atau mengekspor konten snapshot EBS ke format lain. Namun, Anda dapat mengakses data dalam snapshot Anda dengan menggunakan API langsung EBS, atau dengan membuat volume EBS dari snapshot dan mentransfer file.

Mengapa hal ini penting

Jika Anda memiliki data penting bisnis pada volume EBS Anda, kemungkinan besar Anda membuat cadangan dalam bentuk snapshot EBS. Anda dapat mengambil data ini dan menyalinnya ke penyimpanan yang setara secara fungsional.

Opsi 1: Untuk membaca data dari snapshot Anda

Gunakan API langsung EBS.

Opsi 2: Untuk membuat volume dari snapshot dan mentransfer data

1. Gunakan perintah create-volume untuk membuat volume dari snapshot.

```
aws ec2 create-volume \
   --volume-type gp3 \
   --snapshot-id snap-0ad439c50efabb47c \
   --iops 6000 \
   --availability-zone us-east-2b
```

2. Gunakan perintah attach-volume untuk melampirkan volume ke instance.

```
aws ec2 attach-volume \
   --device /dev/sdf \
   --instance-id i-0b1bf24fd4f297ab9 \
   --volume-id vol-1234567890abcdef0
```

3. Hubungkan ke instance Anda dan format dan pasang volume.
4. Transfer file pada volume ke komputer Anda sendiri.

Kuota untuk Amazon EBS

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk masing-masing Layanan AWS. Kecuali dinyatakan lain, setiap kuota adalah Region-specific. Anda dapat meminta peningkatan untuk beberapa kuota dan kuota lainnya tidak dapat ditingkatkan.

Untuk melihat kuota untuk Amazon EBS, buka konsol Kuota Layanan. Di panel navigasi, pilih AWS layanan dan pilih Amazon Elastic Block Store (Amazon EBS). Untuk meminta peningkatan kuota, lihat Meminta Peningkatan Kuota dalam Panduan Pengguna Service Quotas.

Anda Akun AWS memiliki kuota berikut terkait dengan Amazon EBS.

Nama	Default	Dapat disesuaikan	Deskripsi
Snapshot yang diarsipkan per volume	Setiap Wilayah yang didukung: 25	Ya	Jumlah maksimum snapshot yang diarsipkan per volume.
CompleteSnapshot permintaan per akun	Setiap Wilayah yang didukung: 10 per detik	Tidak	Jumlah maksimum CompleteSnapshot permintaan yang diizinkan per akun.
Salinan snapshot bersamaan per Wilayah tujuan	Setiap Wilayah yang didukung: 20	Tidak	Jumlah maksimum salinan snapshot bersamaan ke satu Wilayah tujuan.
Snapshot bersamaan per volume Cold HDD (sc1)	Setiap Wilayah yang didukung: 1	Tidak	Jumlah maksimum snapshot bersamaan per volume Cold HDD (sc1) di Wilayah ini.
Snapshot bersamaan per volume SSD Tujuan Umum (gp2)	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum snapshot bersamaan per volume SSD Tujuan

Nama	Default	Dapat disesu an	Deskripsi
			Umum (gp2) di Wilayah ini.
Snapshot bersamaan per volume SSD Tujuan Umum (gp3)	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum snapshot bersamaan per volume SSD Tujuan Umum (gp3) di Wilayah ini.
Snapshot bersamaan per volume Magnetik (standar)	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum snapshot bersamaan per volume Magnetik (standar) di Wilayah ini.
Snapshot bersamaan per volume Provisioned IOPS SSD (io1)	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum snapshot bersamaan per volume IOPS SSD (io1) yang disediakan di Wilayah ini.
Snapshot bersamaan per volume Provisioned IOPS SSD (io2)	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum snapshot bersamaan per volume IOPS SSD (io2) yang disediakan di Wilayah ini.
Snapshot bersamaan per Volume HDD yang Dioptimalkan Throughput (st1)	Setiap Wilayah yang didukung: 1	Tidak	Jumlah maksimum snapshot bersamaan per volume Throughput Optimized HDD (st1) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Pemulihan snapshot cepat	Setiap Wilayah yang didukung: 5	Ya	Jumlah maksimum snapshot yang dapat diaktifkan untuk pemulihan snapshot cepat di Wilayah ini.
GetSnapshotBlock permintaan per akun	us-east-1:5.000 per detik us-east-2:5.000 per detik us-west-2:5.000 per detik ap-southeast-1:5.000 per detik eu-west-1:5.000 per detik Masing-masing Wilayah lain yang didukung: 1.000 per detik	Ya	Jumlah maksimum GetSnapshotBlock permintaan yang diizinkan per akun.
GetSnapshotBlock permintaan per snapshot	Setiap Wilayah yang didukung: 1.000 per detik	Tidak	Jumlah maksimum GetSnapshotBlock permintaan yang diizinkan per snapshot.

Nama	Default	Dapat disesuaikan	Deskripsi
IOPS untuk volume Provisioned IOPS SSD (io1)	Setiap Wilayah yang didukung: 300.000	Ya	Jumlah agregat maksimum IOPS yang dapat disediakan di seluruh volume Provisioned IOPS SSD (io1) di Wilayah ini.
IOPS untuk volume Provisioned IOPS SSD (io2)	Setiap Wilayah yang didukung: 100.000	Ya	Jumlah agregat maksimum IOPS yang dapat disediakan di seluruh volume Provisioned IOPS SSD (io2) di Wilayah ini.
Modifikasi IOPS untuk volume Provisioned IOPS SSD (io1)	Setiap Wilayah yang didukung: 500.000	Ya	Jumlah agregat maksimum IOPS yang dapat diminta dalam modifikasi volume di seluruh volume Provisioned IOPS SSD (io1) di Wilayah ini.
Modifikasi IOPS untuk volume Provisioned IOPS SSD (io2)	Setiap Wilayah yang didukung: 100.000	Ya	IOPS maksimum saat ini (dari) dan diminta (ke) untuk permintaan modifikasi volume di seluruh volume Provisioned IOPS SSD (io2) di Wilayah ini.
In-progress arsip snapshot per akun	Setiap Wilayah yang didukung: 25	Ya	Jumlah maksimum arsip snapshot dalam proses per akun.

Nama	Default	Dapat disesuaikan	Deskripsi
In-progress snapshot memulihkan dari arsip per akun	Setiap Wilayah yang didukung: 5	Ya	Jumlah maksimum pemulihan snapshot dalam proses dari arsip per akun.
ListChangedBlocks permintaan per akun	Setiap Wilayah yang didukung: 50 per detik	Tidak	Jumlah maksimum ListChangedBlocks permintaan yang diizinkan per akun.
ListSnapshotBlocks permintaan per akun	Setiap Wilayah yang didukung: 50 per detik	Tidak	Jumlah maksimum ListSnapshotBlocks permintaan yang diizinkan per akun.
Operasi volume salinan bersamaan maksimum per akun	Setiap Wilayah yang didukung: 5	Ya	Jumlah maksimum operasi volume penyalinan bersamaan yang diizinkan per akun di wilayah ini.
Tingkat Penyediaan untuk Inisialisasi Volume di seluruh permintaan pembuatan volume bersamaan per Wilayah	Setiap Wilayah yang didukung: 5.000	Ya	Tingkat Penyediaan agregat maksimum untuk Inisialisasi Volume, in MiB/s, yang dapat diminta di seluruh permintaan pembuatan volume bersamaan di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
PutSnapshotBlock permintaan per akun	us-east-1:5.000 per detik us-east-2:5.000 per detik us-west-2:5.000 per detik ap-southeast-1:5.000 per detik eu-west-1:5.000 per detik Masing-masing Wilayah lain yang didukung: 1.000 per detik	Ya	Jumlah maksimum PutSnapshotBlock permintaan yang diizinkan per akun.
PutSnapshotBlock permintaan per snapshot	Setiap Wilayah yang didukung: 1.000 per detik	Tidak	Jumlah maksimum PutSnapshotBlock permintaan yang diizinkan per snapshot.
Snapshot per Wilayah	Setiap Wilayah yang didukung: 100.000	Ya	Jumlah maksimum snapshot per Wilayah.
StartSnapshot snapshot yang tertunda per akun	Setiap Wilayah yang didukung: 100	Tidak	Jumlah maksimum snapshot tertunda per akun yang dapat dibuat menggunakan StartSnapshot API.

Nama	Default	Dapat disesu an	Deskripsi
StartSnapshot permintaan per akun	Setiap Wilayah yang didukung: 10 per detik	Tidak	Jumlah maksimum StartSnapshot permintaa n yang diizinkan per akun.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume Cold-HDD (sc1) dalam TiB	us-timur-1:50 us-timur-2:50 us-barat-1:50 us-west-2:50 ap-northeast-1:50 ap-northeast-2:50 ap-selatan-1:50 ap-Southeast-1:50 ap-Southeast-2:50 ca-sentral-1:50 eu-central-1:50 eu-utara-1:50 eu-west-1:50 eu-west-2:50 eu-west-3:50 sa-east-1:50 Masing-masing Wilayah yang didukung lainnya: 300	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume Cold HDD (sc1) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume SSD Tujuan Umum (gp2) dalam TiB	us-timur-1:50 us-timur-2:50 us-barat-1:50 us-west-2:50 ap-northeast-1:50 ap-northeast-2:50 ap-selatan-1:50 ap-Southeast-1:50 ap-Southeast-2:50 ca-sentral-1:50 eu-central-1:50 eu-utara-1:50 eu-west-1:50 eu-west-2:50 eu-west-3:50 sa-east-1:50 Masing-masing Wilayah yang didukung lainnya: 300	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume SSD Tujuan Umum (gp2) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume SSD Tujuan Umum (gp3) dalam TiB	af-selatan-1:300 ap-east-1:300 ap-east-2:300 ap-northeast-3:300 ap-selatan-2:300 ap-Southeast-4:300 ap-Southeast-5:300 ap-Southeast-6:300 ap-Southeast-7:300 ca-west-1:300 eu-central-2:300 eu-selatan-1:300 eu-selatan-2:300 il-central-1:300 mx-sentral-1:300 Masing-masing Wilayah yang	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume SSD Tujuan Umum (gp3) di Wilayah ini.

Nama	Default	Dapat disesu an	Deskripsi
	didukung lainnya: 50		

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume Magnetik (standar) dalam TiB	us-timur-1:50 us-timur-2:50 us-barat-1:50 us-west-2:50 ap-northeast-1:50 ap-northeast-2:50 ap-selatan-1:50 ap-Southeast-1:50 ap-Southeast-2:50 ca-sentral-1:50 eu-central-1:50 eu-utara-1:50 eu-west-1:50 eu-west-2:50 eu-west-3:50 sa-east-1:50 Masing-masing Wilayah yang didukung lainnya: 300	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume Magnetik (standar) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume Provisioned IOPS SSD (io1) EBS	us-timur-1:50 us-timur-2:50 us-barat-1:50 us-west-2:50 ap-northeast-1:50 ap-northeast-2:50 ap-selatan-1:50 ap-Southeast-1:50 ap-Southeast-2:50 ca-sentral-1:50 eu-central-1:50 eu-utara-1:50 eu-west-1:50 eu-west-2:50 eu-west-3:50 sa-east-1:50 Masing-masing Wilayah yang didukung lainnya: 300	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume Provisioned IOPS SSD (io1) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume Provisioned IOPS SSD (io2) EBS	Setiap Wilayah yang didukung: 20	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume Provisioned IOPS SSD (io2) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Penyimpanan untuk volume HDD yang Dioptimalkan Throughput (st1) dalam TiB	us-timur-1:50 us-timur-2:50 us-barat-1:50 us-west-2:50 ap-northeast-1:50 ap-northeast-2:50 ap-selatan-1:50 ap-Southeast-1:50 ap-Southeast-2:50 ca-sentral-1:50 eu-central-1:50 eu-utara-1:50 eu-west-1:50 eu-west-2:50 eu-west-3:50 sa-east-1:50 Masing-masing Wilayah yang didukung lainnya: 300	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat disediakan di seluruh volume Throughput Optimized HDD (st1) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Modifikasi penyimpanan untuk volume Cold-HDD (sc1) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume Cold HDD (sc1) di Wilayah ini.
Modifikasi penyimpanan untuk volume SSD Tujuan Umum (gp2) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume SSD Tujuan Umum (gp2) di Wilayah ini.
Modifikasi penyimpanan untuk volume SSD Tujuan Umum (gp3) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume SSD Tujuan Umum (gp3) di Wilayah ini.
Modifikasi penyimpanan untuk volume Magnetik (standar) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume Magnetik (standar) di Wilayah ini.

Nama	Default	Dapat disesuaikan	Deskripsi
Modifikasi penyimpanan untuk volume Provisioned IOPS SSD (io1) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume Provisioned IOPS SSD (io1) di Wilayah ini.
Modifikasi penyimpanan untuk volume Provisioned IOPS SSD (io2) dalam TiB	Setiap Wilayah yang didukung: 20	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume Provisioned IOPS SSD (io2) di Wilayah ini.
Modifikasi penyimpanan untuk volume HDD yang Dioptimalkan Throughput (st1) dalam TiB	Setiap Wilayah yang didukung: 500	Ya	Jumlah penyimpanan agregat maksimum, dalam TiB, yang dapat diminta dalam modifikasi volume di seluruh volume Throughput Optimized HDD (st1) di Wilayah ini.
Time-based throughput salinan snapshot per Wilayah tujuan	Setiap Wilayah yang didukung: 2.000	Ya	Throughput tingkat akun maksimum, dalam MiB/sec, untuk operasi penyalinan snapshot berbasis waktu per Wilayah tujuan.

Pertimbangan-pertimbangan

- Kuota Anda dapat berubah seiring waktu. Amazon EBS terus memantau penyimpanan yang disediakan dan penggunaan IOPS di setiap Wilayah dan mungkin secara otomatis meningkatkan kuota Anda, berdasarkan Per-wilayah, berdasarkan penggunaan Anda. Meskipun Amazon EBS dapat secara otomatis meningkatkan kuota berdasarkan penggunaan Anda, Anda dapat meminta kenaikan kuota jika diperlukan. Misalnya, jika Anda berencana untuk menggunakan lebih banyak gp3 penyimpanan di AS Timur (Virginia Utara) daripada kuota Anda saat ini, Anda dapat meminta peningkatan kuota untuk jenis volume tersebut di Wilayah tersebut sebelum penggunaan yang direncanakan.
- Kuota untuk salinan snapshot Serentak per Wilayah tujuan tidak dapat disesuaikan menggunakan Kuota Layanan. Namun, Anda dapat meminta kenaikan kuota ini dengan menghubungi AWS Dukungan.
- Modi fikasi IOPS dan ku ota modifikasi Penyimpanan berlaku untuk nilai agregat saat ini (untuk ukuran atau IOPS, tergantung pada kuota) volume yang dapat mengalami modifikasi secara bersamaan. Anda dapat membuat permintaan modifikasi bersamaan untuk volume yang telah menggabungkan nilai saat ini (untuk ukuran atau IOPS) hingga kuota. Misalnya, jika kuota modifikasi IOPS untuk volume Provisioned IOPS SSD (io1) adalah`50,000`, Anda dapat membuat permintaan modifikasi IOPS bersamaan untuk sejumlah `io1` volume selama gabungan IOPS saat ini sama dengan atau kurang dari. `50,000` Jika Anda memiliki tiga `io1` volume yang disediakan dengan masing-masing `20,000` IOPS, Anda dapat meminta modifikasi IOPS untuk dua volume secara bersamaan (). `20,000 * 2 < 50,000` Jika Anda mengirimkan permintaan modifikasi IOPS bersamaan untuk volume ketiga, Anda melebihi kuota Anda dan permintaan tersebut gagal ()`20,000 * 3 > 50,000`.
- Amazon EBS memiliki batas yang tidak dapat disesuaikan berikut untuk jumlah volume EBS per permintaan peluncuran instans.
  - `2500`—`us-east-1`,`us-west-2`,`eu-west-1`, dan `ap-northeast-1`
  - `500`— Semua Wilayah lainnya

  Batas ini berlaku untuk permintaan peluncuran instans yang Anda buat, dan permintaan peluncuran instance yang dibuat oleh AWS layanan, seperti Amazon EMR, atas nama Anda. Jika permintaan peluncuran instans gagal karena melebihi batas ini, sesuaikan konfigurasi volume EBS dalam permintaan peluncuran untuk mengurangi jumlah volume di bawah batas. Atau, bekerjasama dengan manajer akun teknis (TAM) Anda untuk menjelajahi opsi lain untuk meluncurkan cluster Anda.

Riwayat dokumen untuk Panduan Pengguna Amazon EBS

Tabel berikut menjelaskan rilis dokumentasi untuk Amazon EBS.

Perubahan	Deskripsi	Tanggal
Pembaruan modifikasi Volume Elastic	Anda sekarang dapat memulai modifikasi Volume Elastic baru segera setelah yang sebelumnya selesai, hingga empat kali dalam periode 24 jam.	Januari 15, 2026
Recycle Bin untuk volume EBS	Recycle Bin memungkinkan Anda melindungi volume Amazon EBS dari penghapus an yang tidak disengaja.	20 November 2025
Menyalin volume EBS	Anda dapat membuat salinan instan, point-in-time volume Amazon EBS dalam Availabil ity Zone yang sama.	Oktober 14, 2025
pembaruan volume gp3	Anda sekarang dapat menyediakan volume General Purpose SSD (gp3) dengan ukuran hingga 64 TiB, IOPS hingga 80.000, dan throughput hingga 2.000. MiB/s	September 26, 2025
Pembaruan panduan latensi untuk volume io2	`io2`Volume Block Express dirancang untuk memberika n latensi rata-rata di bawah 500 mikrodetik untuk operasi 16KiB I/O . `io2` Volume Block Express juga memberikan latensi outlier yang lebih baik	September 26, 2025

	dibandingkan dengan volume Tujuan Umum, mengurangi frekuensi I/Os melebihi 800 mikrodetik lebih dari 10 kali lipat.	
Injeksi Latensi	Gunakan tindakan Injeksi AWS FIS latensi untuk mensimulasikan peningkat an I/O latensi pada volume Amazon EBS Anda.	September 16, 2025
`io2`Perluasan wilayah	Volume IOPS SSD (`io2`) yang disediakan sekarang tersedia di semua komersial dan AWS GovCloud (US) Wilayah.	Juli 22, 2025
Snapshot lokal di Zona Lokal	Anda sekarang dapat membuat snapshot lokal di Zona Lokal.	Mei 15, 2025
Tarif Penyediaan Amazon EBS untuk Inisialisasi Volume	Anda sekarang dapat menentukan tingkat inisialis asi volume saat membuat volume Amazon EBS untuk memastikan bahwa volume tersebut sepenuhnya diinisial isasi dalam jumlah waktu yang dapat diprediksi.	6 Mei 2025
Titik akhir VPC Manajer Siklus Hidup Data Amazon	Anda sekarang dapat membuat koneksi pribadi antara VPC dan Amazon Data Lifecycle Manager dengan membuat antarmuka titik akhir VPC.	Februari 28, 2025

Time-based Salinan AMI	Anda sekarang dapat meminta durasi penyelesaian untuk EBS-backed operasi penyalina n AMI untuk memastikan bahwa salinan AMI selesai dalam jangka waktu tertentu.	Februari 25, 2025
Ukuran snapshot penuh	Anda sekarang dapat melihat ukuran penuh snapshot Amazon EBS menggunak an konsol Amazon EC2 dan. AWS CLI	Februari 11, 2025
Dukungan IPv6 Manajer Siklus Hidup Data Amazon	Amazon Data Lifecycle Manager kini menyediakan titik akhir dual-stack yang mendukung lalu lintas IPv4 dan IPv6.	Februari 7, 2025
Dukungan Recycle Bin IPv6	Recycle Bin sekarang menyediakan titik akhir dual-stack yang mendukung lalu lintas IPv4 dan IPv6.	Desember 19, 2024
Snapshot lokal di Zona Lokal Khusus	Anda sekarang dapat membuat snapshot lokal di Zona Lokal Khusus.	Desember 16, 2024
AWSDataLifecycleMa nagerServiceRole AWS kebijakan terkelola diperbarui	Kebijakan AWSDataLi fecycleManagerServiceRole AWS terkelola telah diperbaru i untuk menyertakan izin untuk `ec2:DescribeAvaila bilityZones` tindakan tersebut.	Desember 16, 2024

Kebijakan deklaratif untuk memblokir akses publik untuk snapshot EBS	Anda sekarang dapat menggunakan kebijakan deklaratif untuk menerapkan pengaturan tingkat akun untuk memblokir akses publik untuk snapshot di beberapa Wilayah dan akun secara bersamaan. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.	Desember 1, 2024
Time-based salinan snapshot	Anda sekarang dapat meminta durasi penyelesaian untuk operasi penyalinan snapshot untuk memastikan bahwa salinan snapshot selesai dalam jangka waktu tertentu.	November 26, 2024
Tag pengecualian untuk Recycle Bin	Anda sekarang dapat menambahkan tag pengecualian ke aturan Region-level penyimpanan untuk mengecualikan sumber daya yang memiliki tag tertentu.	November 19, 2024
CloudFormation dukungan untuk Recycle Bin	Anda sekarang dapat membuat dan mengelola aturan retensi Recycle Bin menggunakan AWS CloudFormation.	November 18, 2024

Statistik kinerja terperinci Amazon EBS	Perangkat blok Amazon EBS NVMe menjual statistik I/O kinerja resolusi tinggi real-time untuk volume Amazon EBS yang terpasang pada instans Amazon EC2. Nitro-based	November 12, 2024
CloudWatch Metrik baru untuk volume Amazon EBS	Anda sekarang dapat menggunakan CloudWatch metrik `VolumeAvgReadLatency` `VolumeAvgWriteLatency` ,`VolumeIOPSExceededCheck` ,, dan `VolumeThroughputExceededCheck` Amazon untuk memantau kinerja volume.	Oktober 30, 2024
Aktifkan kebijakan default Amazon Data Lifecycle Manager di seluruh akun	Anda dapat menggunakannya CloudFormation StackSets untuk mengaktifkan kebijakan default Amazon Data Lifecycle Manager di seluruh AWS organisasi atau di seluruh AWS akun tertentu.	April 26, 2024
AWSDataLifecycleManagerSSMFullAccess AWS kebijakan yang dikelola	Memperbarui kebijakan untuk mendukung snapshot yang konsisten dengan aplikasi untuk SAP HANA menggunakan dokumen SSM `AWSSystemsManagerSAP-CreateDLMSnapshotForSAPHANA` .	17 November 2023

VolumeStalledIOCheck metrik	Anda dapat menggunakan metrik `VolumeStalledIOCheck` untuk memeriksa apakah volume telah berhasil atau gagal melewati pemeriksaan IO yang macet di menit terakhir.	16 November 2023
Kebijakan default Amazon Data Lifecycle Manager	Anda sekarang dapat membuat kebijakan default Amazon Data Lifecycle Manager untuk snapshot EBS dan EBS-backed AMI untuk membuat cadangan semua volume dan instans di Wilayah.	16 November 2023
Kunci snapshot Amazon EBS	Anda dapat mengunci snapshot Amazon EBS untuk melindunginya dari penghapusan yang tidak disengaja atau berbahaya, atau menyimpannya dalam format WORM untuk durasi tertentu.	15 November 2023
Blokir akses publik untuk snapshot	Anda sekarang dapat menggunakan blokir akses publik untuk snapshot guna mencegah berbagi snapshot secara publik.	9 November 2023

Praskrip dan pascaskrip Amazon Data Lifecycle Manager	Sekarang Anda dapat menggunakan praskrip dan pascaskrip dalam kebijakan snapshot Amazon Data Lifecycle Manager untuk mengotomatisasi siklus hidup snapshot yang konsisten dengan aplikasi.	7 November 2023
Reservasi NVMe	Multi-Attach `io2`volume yang diaktifkan mendukung reservasi NVMe, yang merupakan seperangkat protokol pagar penyimpanan standar industri.	18 September 2023
Pengujian kesalahan pada Amazon EBS	Gunakan AWS FIS untuk berhenti sementara I/O antara volume EBS dan instans yang dilampirkan untuk menguji bagaimana beban kerja Anda menangani I/O interupsi.	27 Januari 2023
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
Kunci syarat untuk Keranjang Sampah	Anda dapat menggunakan kunci syarat `rbin:Request/ResourceType` dan `rbin:Attribute/ResourceType` 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
Keranjang Sampah untuk AMI	Keranjang Sampah memungkinkan Anda memulihkan AMI yang terhapus secara tidak sengaja.	3 Februari 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
Arsip Snapshot Amazon EBS	Arsip Snapshot Amazon EBS adalah tingkat penyimpan an baru yang dapat Anda gunakan untuk penyimpanan berbiaya rendah dan jangka panjang dari snapshot yang jarang diakses.	29 November 2021
Dukungan pengusangan AMI untuk Amazon Data Lifecycle Manager	Kebijakan EBS-backed AMI Amazon Data Lifecycle Manager dapat menghentikan AMI. Kebijakan AWSDataLi fecycleManagerServiceRoleFo rAMIManagement AWS terkelola telah diperbarui untuk mendukung fitur ini.	23 Agustus 2021

CloudWatch metrik untuk Amazon Data Lifecycle Manager	Anda dapat memantau kebijakan Amazon Data Lifecycle Manager menggunakan Amazon CloudWatch.	28 Juli 2021
CloudTrail peristiwa data untuk API langsung EBS	PutSnapshotBlock API ListSnapshotBlocks , ListChangedBlocks GetSnapshotBlock,, dan dapat dicatat peristiwa data di CloudTrail.	27 Juli 2021
Volume `io2` Block Express	`io2`Volume Block Express sekarang tersedia secara umum.	19 Juli 2021
Snapshot lokal Amazon EBS di Outposts	Anda sekarang dapat menggunakan snapshot lokal Amazon EBS di Outposts untuk menyimpan snapshot volume Outpost secara lokal di Amazon S3 pada itu sendiri. Outpost	4 Februari 2021
Multi-Attach dukungan untuk volume io2	Anda sekarang dapat mengaktifkan volume IOPS SSD (`io2`) yang disediakan untuk Amazon EBS. Multi-Attach	18 Desember 2020
Amazon Data Lifecycle Manager	Gunakan Amazon Data Lifecycle Manager untuk mengotomatiskan proses berbagi snapshot dan menyalinnya di seluruh AWS akun.	17 Desember 2020

Volume gp3	Tipe volume SSD Tujuan Umum Amazon EBS baru. Anda dapat menentukan IOPS yang tersedia dan throughput saat Anda membuat atau memodifikasi volume.	1 Desember 2020
Ukuran volume HDD dengan throughput yang dioptimalkan dan HDD Cold	Volume HDD dengan throughput yang dioptimalkan (`st1`) dan HDD Cold (`sc1`) dapat mempunyai ukuran berkisar dari 125 GiB hingga 16 TiB.	30 November 2020
Amazon Data Lifecycle Manager	Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pembuatan, penyimpanan, dan penghapusan EBS-backed AMI.	9 November 2020
Amazon Data Lifecycle Manager	Kebijakan Amazon Data Lifecycle Manager dapat dikonfigurasi dengan hingga empat jadwal.	17 September 2020
Volume IOPS SSD (io2) yang disediakan untuk Amazon EBS	Volume (`io2`) IOPS SSD yang disediakan dirancang untuk memberikan ketahanan volume 99,999 persen dengan AFR tidak lebih dari 0,001 persen.	24 Agustus 2020
Pemulihan snapshot cepat	Anda dapat mengaktifkan pemulihan snapshot cepat untuk snapshot yang dibagikan dengan Anda.	21 Juli 2020

Amazon EBS Multi-Attach	Anda sekarang dapat melampirkan satu volume Provisioned IOPS SSD (io1) ke hingga 16 Nitro-based instans yang berada di Zona Ketersediaan yang sama.	14 Februari 2020
Pemulihan snapshot cepat Amazon EBS	Anda dapat mengaktifkan pemulihan snapshot cepat pada snapshot EBS untuk memastikan bahwa volume EBS yang dibuat dari snapshot sepenuhnya diinisialisasi saat pembuatan dan langsung memberikan semua kinerja yang disediakan.	20 November 2019
Snapshot multi-volume Amazon EBS	Anda dapat mengambil snapshot titik waktu yang tepat, terkoordinasi data, dan crash-consistent di banyak volume EBS yang terlampir ke instans EC2.	29 Mei 2019
Enkripsi Amazon EBS secara default	Setelah Anda mengaktif kan enkripsi secara default di suatu Wilayah, semua volume EBS baru yang dibuat di Wilayah tersebut akan dienkripsi dengan kunci KMS default untuk enkripsi EBS.	23 Mei 2019

Otomatisasi siklus hidup snapshot	Anda dapat menggunakan Amazon Data Lifecycle Manager guna mengotomatisasi pembuatan dan penghapusan snapshot untuk volume EBS Anda.	12 Juli 2018
Melakukan modifikasi pada volume EBS terlampir	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
Salin snapshot Amazon EBS terenkripsi di antara Akun AWS	Anda sekarang dapat menyalin snapshot EBS terenkripsi di antara. Akun AWS	Juni 21, 2016
Throughput Jenis volume HDD dan HDD Dingin yang dioptimalkan	Anda sekarang dapat membuat volume HDD Throughput yang Dioptimalkan (st1) dan HDD Cold (sc1).	19 April 2016

Jenis volume SSD Tujuan Umum	Volume SSD Tujuan Umum menawarkan penyimpanan hemat biaya yang ideal untuk berbagai beban kerja. Volume ini memberikan latensi milidetik satu digit, kemampuan untuk meledak hingga 3.000 IOPS untuk jangka waktu yang lama, dan kinerja dasar 3. IOPS/GiB Ukuran volume SSD Tujuan Umum dapat bervariasi mulai 1 GiB hingga 1 TiB.	16 Juni 2014
Enkripsi Amazon EBS	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
Salinan snapshot inkremental	Anda sekarang dapat menjalankan salinan snapshot inkremental.	11 Juni 2013

Salinan snapshot EBS	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

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.