aws

Panduan Pengguna

Amazon S3 on Outposts

Versi API 2006-03-01

Amazon S3 on Outposts: Panduan Pengguna

Table of Contents

Apa itu Amazon S3 di Outposts?

AWS Outposts adalah layanan yang dikelola sepenuhnya yang menawarkan AWS infrastruktur, AWS layanan APIs, dan alat yang sama ke hampir semua pusat data, ruang co-lokasi, atau fasilitas lokal untuk pengalaman hybrid yang benar-benar konsisten. AWS Outposts sangat ideal untuk beban kerja yang memerlukan akses latensi rendah ke sistem lokal, pemrosesan data lokal, residensi data, dan migrasi aplikasi dengan saling ketergantungan sistem lokal. Untuk informasi selengkapnya, lihat Apa itu AWS Outposts? dalam AWS Outposts Panduan Pengguna.

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di Outposts Anda serta dengan mudah menyimpan dan mengambil objek on-premise. S3 on Outposts menyediakan kelas penyimpanan baru0UTP0STS, yang menggunakan Amazon APIs S3 dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di Outposts Anda. Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC).

Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outposts seperti yang Anda lakukan di Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API.

Cara kerja S3 di Outposts

S3 di Outposts adalah layanan penyimpanan objek yang menyimpan data sebagai objek dalam bucket di Outpost Anda. Objek adalah file data dan metadata opsional apa pun yang menjelaskan file tersebut. Bucket adalah kontainer untuk objek.

Untuk menyimpan data Anda di S3 di Outposts, Anda terlebih dahulu membuat bucket. Saat membuat bucket, Anda menentukan nama bucket dan Outpost yang akan menampung bucket. Untuk

mengakses bucket S3 di Outposts dan mengoperasikan objek, Anda selanjutnya membuat dan mengonfigurasikan titik akses. Anda juga harus membuat titik akhir untuk merutekan permintaan ke titik akses Anda.

Titik akses menyederhanakan akses data untuk aplikasi apa pun Layanan AWS atau pelanggan yang menyimpan data di S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket serta dapat digunakan untuk melakukan operasi objek, seperti `GetObject` dan `PutObject`. Setiap titik akses memiliki izin dan kendali jaringan yang berbeda.

Anda dapat membuat dan mengelola S3 di bucket Outposts, titik akses, dan titik akhir dengan menggunakan AWS Management Console,, atau REST AWS CLI API AWS SDKs. Untuk mengunggah dan mengelola objek di bucket S3 di Outposts, Anda dapat menggunakan AWS CLI, AWS SDKs, atau REST API.

Wilayah

Selama AWS Outposts penyediaan, Anda atau AWS membuat koneksi tautan layanan yang menghubungkan Pos Luar Anda kembali ke Wilayah Outposts pilihan Anda atau Wilayah AWS Outposts untuk operasi bucket dan telemetri. Outposts mengandalkan konektivitas ke Wilayah AWS induk. Rak Outposts tidak dirancang untuk operasi atau lingkungan yang tidak terhubung dengan konektivitas terbatas atau tanpa konektivitas. Untuk informasi selengkapnya, lihat Konektivitas Outposts ke Wilayah AWS di AWS Outposts Panduan Pengguna.

Bucket

Bucket adalah kontainer untuk objek yang disimpan dalam S3 di Outposts. Anda dapat menyimpan berapa pun jumlah objek dalam bucket dan dapat memiliki hingga 100 bucket per akun per Outpost.

Saat Anda membuat bucket, Anda memasukkan nama bucket dan memilih Outpost tempat bucket akan berada. Setelah Anda membuat bucket, Anda tidak dapat mengubah nama bucket atau memindahkan bucket ke Outpost lain. Nama bucket harus mengikuti aturan penamaan bucket Amazon S3. Di S3 on Outposts, nama bucket unik untuk Outpost dan. Akun AWS Bucket S3 di Outposts memerlukan `outpost-id`, `account-id`, dan nama bucket untuk mengidentifikasinya.

Contoh berikut menunjukkan format Amazon Resource Name (ARN) untuk bucket S3 di Outposts. ARN terdiri atas Wilayah asal Outposts Anda, akun Outpost Anda, ID Outposts, dan nama bucket.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/bucket/bucket-name
```

Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan ARN titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN titik akses untuk S3 di Outposts, yang mencakup `outpost-id`, `account-id`, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi selengkapnya tentang bucket, lihat Bekerja dengan S3 di bucket Outposts.

Objek

Objek adalah entitas dasar yang disimpan di S3 di Outposts. Objek terdiri atas data objek dan metadata. Metadata adalah serangkaian pasangan nilai-nama yang menjelaskan objek. Pasangan ini mencakup beberapa metadata default, seperti tanggal terakhir diubah, dan metadata HTTP standar, seperti `Content-Type`. Anda juga dapat menentukan metadata kustom pada saat objek disimpan. Objek diidentifikasi secara unik dalam bucket dengan kunci (atau nama).

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Kunci

Kunci objek (atau nama kunci) adalah pengidentifikasi unik untuk objek dalam bucket. Setiap objek dalam bucket memiliki persis satu kunci. Kombinasi bucket dan kunci objek secara unik mengidentifikasi setiap objek.

Contoh berikut menunjukkan format ARN untuk S3 pada objek Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar ditempatkan, Akun AWS ID, ID Outpost, nama bucket, dan kunci objek:

```
arn:aws:s3-outposts:us-west-2:123456789012:outpost/ op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket1/object/myobject
```

Untuk informasi selengkapnya tentang kunci objek, lihat Bekerja dengan S3 di objek Outposts.

Penentuan Versi S3

Anda dapat menggunakan Penentuan Versi S3 di bucket Outposts untuk menyimpan beberapa varian objek dalam bucket yang sama. Dengan Penentuan Versi S3, Anda dapat menyimpan, mengambil, dan memulihkan setiap versi dari setiap objek yang disimpan dalam bucket Anda. Penentuan Versi S3 membantu Anda memulihkan dari tindakan pengguna yang tidak diinginkan dan kegagalan aplikasi.

Untuk informasi selengkapnya, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.

ID Versi

Saat Anda mengaktifkan Penentuan Versi S3 dalam bucket, S3 di Outposts menghasilkan ID versi unik untuk setiap objek yang ditambahkan ke bucket. Objek yang sudah ada di bucket pada saat Anda mengaktifkan Penentuan Versi memiliki ID versi `null`. Jika Anda memodifikasi objek ini (atau lainnya) dengan operasi lain PutObject, seperti, objek baru mendapatkan ID versi unik.

Untuk informasi selengkapnya, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.

Kelas penyimpanan dan enkripsi

S3 di Outposts menyediakan kelas penyimpanan baru, S3 Outposts (`OUTPOSTS`). Kelas penyimpanan S3 Outposts hanya tersedia untuk objek yang disimpan dalam bucket di AWS Outposts. Jika Anda mencoba menggunakan kelas penyimpanan S3 lainnya dengan S3 pada Outposts, S3 di Outposts mengembalikan kesalahan `InvalidStorageClass` tersebut.

Secara default, objek yang disimpan dalam kelas penyimpanan S3 Outposts (`OUTPOSTS`) disimpan menggunakan enkripsi di sisi server dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Untuk informasi selengkapnya, lihat Enkripsi data dalam S3 di Outposts.

Kebijakan bucket

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya.

Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB.

Kebijakan bucket menggunakan bahasa kebijakan IAM berbasis JSON yang standar di seluruh AWS. Anda dapat menggunakan kebijakan bucket untuk menambah atau menolak izin objek dalam bucket. Kebijakan bucket mengizinkan atau menolak permintaan berdasarkan elemen dalam kebijakan. Elemen ini dapat mencakup pemohon, tindakan S3 pada Outposts, sumber daya, dan aspek atau ketentuan permintaan (misalnya, alamat IP yang digunakan untuk membuat permintaan). Misalnya, Anda dapat membuat kebijakan bucket yang memberikan izin lintas akun untuk mengunggah objek ke bucket S3 di Outposts sekaligus memastikan pemilik bucket memiliki kendali penuh atas objek yang diunggah.

Dalam kebijakan bucket, Anda dapat menggunakan karakter wildcard (`*`) di ARNs dan nilai lainnya untuk memberikan izin ke subset objek. Misalnya, Anda dapat mengendalikan akses ke kelompok objek yang dimulai dengan prefiks umum atau diakhiri dengan ekstensi tertentu, seperti `.html`.

Titik akses S3 di Outposts

Titik akses S3 di Outposts diberi nama titik akhir jaringan dengan kebijakan akses khusus yang menjelaskan cara data dapat diakses menggunakan titik akhir tersebut. Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama dalam S3 di Outposts. Titik akses dilampirkan ke bucket yang dapat Anda gunakan untuk melakukan operasi objek S3, seperti `GetObject` dan `PutObject`.

Saat menentukan bucket untuk operasi objek, Anda menggunakan ARN titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Titik akses memiliki izin dan kendali jaringan yang berbeda yang diterapkan S3 di Outposts untuk setiap permintaan yang dibuat melalui titik akses tersebut. Setiap titik akses memberlakukan kebijakan titik akses khusus yang bekerja bersama kebijakan bucket yang melekat pada bucket dasar.

Untuk informasi selengkapnya, lihat Mengakses S3 Anda di bucket dan objek Outposts.

Fitur S3 di Outposts

Manajemen akses

S3 di Outposts menyediakan fitur untuk mengaudit dan mengelola akses ke bucket dan objek Anda. Secara default, S3 pada bucket Outposts dan objek di dalamnya bersifat pribadi. Anda hanya memiliki akses ke sumber daya S3 di Outposts yang Anda buat.

Untuk memberikan izin sumber daya terperinci yang mendukung kasus penggunaan spesifik Anda atau untuk mengaudit izin sumber daya S3 di Outposts, Anda dapat menggunakan fitur berikut.

- S3 Blokir Akses Publik–Memblokir akses publik ke bucket dan objek. Untuk bucket di Outposts, Blokir Akses Publik selalu diaktifkan secara default.
- AWS Identity and Access Management (IAM) — IAM adalah layanan web yang membantu Anda mengontrol akses ke AWS sumber daya dengan aman, termasuk sumber daya S3 on Outposts Anda. Dengan IAM, Anda dapat mengelola izin secara terpusat yang mengendalikan sumber daya AWS yang dapat diakses pengguna. Anda menggunakan IAM untuk mengontrol siapa yang diautentikasi (masuk) dan diotorisasi (memiliki izin) untuk menggunakan sumber daya.
- Titik akses S3 di Outposts–Mengelola akses data untuk set data bersama di S3 di Outposts. Titik akses diberi nama titik akhir jaringan dengan kebijakan akses khusus. Titik akses melekat ke bucket dan dapat digunakan untuk melakukan operasi objek, seperti `GetObject` dan `PutObject`.
- Kebijakan bucket–Gunakan bahasa kebijakan berbasis IAM untuk mengonfigurasi izin berbasis sumber daya untuk bucket S3 dan objek di dalamnya.
- AWS Resource Access Manager (AWS RAM) — Bagikan kapasitas S3 on Outposts Anda dengan aman di seluruh Akun AWS, di dalam organisasi atau unit organisasi Anda () di. OUs AWS Organizations

Pencatatan dan pemantauan penyimpanan

S3 di Outposts menyediakan alat bantu pencatatan dan pemantauan yang dapat Anda gunakan untuk memantau dan mengendalikan cara sumber daya S3 di Outposts Anda digunakan. Untuk informasi selengkapnya, lihat Alat pemantauan.

- CloudWatch Metrik Amazon untuk S3 di Outposts — Lacak kesehatan operasional sumber daya Anda dan pahami ketersediaan kapasitas Anda.

- CloudWatch Acara Amazon Events untuk S3 di Outposts - Buat aturan untuk setiap peristiwa S3 pada Outposts API untuk menerima pemberitahuan melalui CloudWatch semua target Acara yang didukung, termasuk Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS), dan. AWS Lambda
- AWS CloudTrail log untuk S3 di Outposts — Rekam tindakan yang diambil oleh pengguna, peran, atau Layanan AWS di S3 di Outposts. CloudTrail log memberi Anda pelacakan API terperinci untuk operasi tingkat ember dan tingkat objek S3.

Konsistensi kuat

S3 di Outposts memberikan konsistensi read-after-write yang kuat untuk permintaan PUT dan DELETE objek di bucket S3 on Outposts Anda. Wilayah AWS Perilaku ini berlaku untuk penulisan objek baru dan permintaan LETAKKAN yang menimpa objek yang ada serta untuk HAPUS permintaan. Selain itu, tag objek dan metadata objek S3 di Outposts (misalnya, objek HEAD) sangat konsisten. Untuk informasi selengkapnya, lihat model konsistensi data Amazon S3 di Panduan Pengguna Amazon S3.

Layanan terkait

Setelah Anda memuat data Anda ke S3 di Outposts, Anda dapat menggunakannya dengan Layanan AWS yang lain. Berikut adalah layanan yang mungkin paling sering Anda gunakan:

- Amazon Elastic Compute Cloud (Amazon EC2) — Menyediakan kapasitas komputasi yang aman dan dapat diskalakan di. AWS Cloud Menggunakan Amazon EC2 mengurangi kebutuhan Anda untuk berinvestasi dalam perangkat keras di muka, sehingga Anda dapat mengembangkan dan menerapkan aplikasi lebih cepat. Anda dapat menggunakan Amazon EC2 untuk meluncurkan server virtual sebanyak atau sesedikit yang Anda butuhkan, mengonfigurasi keamanan dan jaringan, dan mengelola penyimpanan.
- Amazon Elastic Block Store (Amazon EBS) di Outposts–Gunakan snapshot lokal Amazon EBS pada Outposts untuk menyimpan snapshot volume pada Outpost secara lokal dalam S3 di Outposts.
- Amazon Relational Database Service (Amazon RDS) di Outposts–Gunakan cadangan lokal Amazon RDS untuk menyimpan cadangan Amazon RDS Anda secara lokal di Outpost Anda.
- AWS DataSync— Otomatiskan transfer data antara Outposts Anda dan Wilayah AWS, memilih apa yang akan ditransfer, kapan harus mentransfer, dan berapa banyak bandwidth jaringan yang akan digunakan. S3 di Outposts terintegrasi dengan. AWS DataSync Untuk aplikasi on-premise

yang memerlukan pengolahan lokal dengan throughput tinggi, S3 di Outposts menyediakan penyimpanan objek on-premise untuk meminimalkan transfer data dan buffer dari variasi jaringan, sekaligus memberi Anda kemampuan untuk mentransfer data antara Outposts dan Wilayah AWS dengan mudah.

Mengakses S3 di Outposts

Anda dapat bekerja dengan S3 di Outposts dengan salah satu cara berikut ini:

AWS Management Console

Konsol adalah antarmuka pengguna berbasis web untuk mengelola S3 di Outposts dan sumber daya AWS . Jika Anda telah mendaftar untuk Akun AWS, Anda dapat mengakses S3 di Outposts dengan masuk ke AWS Management Console dan memilih S3 dari AWS Management Console halaman beranda. Lalu pilih bucket Outposts dari panel navigasi kiri.

AWS Command Line Interface

Anda dapat menggunakan alat baris AWS perintah untuk mengeluarkan perintah atau membangun skrip di baris perintah sistem Anda untuk melakukan tugas AWS (termasuk S3).

The AWS Command Line Interface (AWS CLI) menyediakan perintah untuk satu set yang luas Layanan AWS. AWS CLI Ini didukung di Windows, macOS, dan Linux. Untuk memulai, lihat Panduan Pengguna AWS Command Line Interface. Untuk informasi selengkapnya tentang perintah yang dapat Anda gunakan dengan S3 di Outposts, lihat s3api, s3control, dan s3outposts dalam Referensi Perintah AWS CLI .

AWS SDKs

AWS menyediakan SDKs (perangkat pengembangan perangkat lunak) yang terdiri dari pustaka dan kode sampel untuk berbagai bahasa dan platform pemrograman (Java, Python, Ruby, .NET, iOS, Android, dan sebagainya). AWS SDKs Menyediakan cara mudah untuk membuat akses terprogram ke S3 di Outposts dan. AWS Karena S3 di Outposts menggunakan SDKs yang sama dengan Amazon S3, S3 on Outposts memberikan pengalaman yang konsisten menggunakan S3, otomatisasi, dan alat yang sama. APIs

S3 di Outposts adalah layanan REST. Anda dapat mengirim permintaan ke S3 di Outposts menggunakan pustaka SDK AWS , yang membungkus API REST yang mendasarinya dan

menyederhanakan tugas pemrograman Anda. Misalnya, SDKs mengurus tugas-tugas seperti menghitung tanda tangan, menandatangani permintaan secara kriptografis, mengelola kesalahan, dan mencoba ulang permintaan secara otomatis. Untuk informasi tentang AWS SDKs, termasuk cara mengunduh dan menginstalnya, lihat Alat untuk Dibangun AWS.

Membayar untuk S3 di Outposts

Anda dapat membeli berbagai konfigurasi AWS Outposts rak yang menampilkan kombinasi jenis EC2 instans Amazon, volume (SSD) solid state drive (SSD) Amazon EBS General Purpose (gp2), dan S3 di Outposts. Harga termasuk pengiriman, instalasi, pemeliharaan layanan infrastruktur, serta patch dan peningkatan perangkat lunak.

Untuk informasi selengkapnya lihat AWS Outposts harga rak.

Langkah selanjutnya

Untuk informasi selengkapnya tentang bekerja dengan S3 di Outposts, lihat topik berikut:

- Menyiapkan Outpost Anda
- Bagaimana Amazon S3 di Outposts berbeda dari Amazon S3?
- Memulai dengan Amazon S3 di Outposts
- Jaringan untuk S3 di Outposts
- Bekerja dengan S3 di bucket Outposts
- Bekerja dengan S3 di objek Outposts
- Keamanan di S3 di Outposts
- Mengelola penyimpanan S3 di Outposts
- Mengembangkan dengan Amazon S3 di Outposts

Menyiapkan Outpost Anda

Untuk memulai dengan Amazon S3 di Outpost, Anda memerlukan Outpost dengan kapasitas Amazon S3 yang diterapkan di fasilitas Anda. Untuk informasi tentang opsi pemesanan kapasitas Outpost dan S3, lihat. AWS Outposts Untuk memeriksa apakah Outposts Anda memiliki kapasitas S3, Anda dapat menggunakan panggilan API ListOutpostsWithS3. Untuk spesifikasi dan untuk melihat bagaimana S3 di Outposts berbeda dari Amazon S3, lihat Bagaimana Amazon S3 di Outposts berbeda dari Amazon S3?

Untuk informasi selengkapnya, lihat topik berikut.

Topik

Pesan Pos Luar baru

Jika Anda perlu memesan Outpost baru dengan kapasitas S3, lihat harga AWS Outposts rak untuk memahami opsi kapasitas untuk Amazon Elastic Compute Cloud (Amazon) EC2, Amazon Elastic Block Store (Amazon EBS), dan Amazon S3.

Setelah memilih konfigurasi Anda, ikuti langkah dalam Buat Outpost dan pesan kapasitas Outpost dalam AWS Outposts Panduan Pengguna.

Bagaimana Amazon S3 di Outposts berbeda dari Amazon S3?

Amazon S3 di Outposts mengirimkan penyimpanan objek ke lingkungan lokal Anda. AWS Outposts Menggunakan S3 di Outposts membantu Anda memenuhi kebutuhan pemrosesan lokal, residensi data, dan kinerja yang menuntut dengan menjaga data tetap dekat dengan aplikasi on-premise. Karena menggunakan Amazon S3 APIs dan fitur, S3 on Outposts memudahkan untuk menyimpan, mengamankan, menandai, melaporkan, dan mengontrol akses ke data di Outposts Anda dan memperluas AWS infrastruktur ke fasilitas lokal Anda untuk pengalaman hybrid yang konsisten.

Untuk informasi selengkapnya tentang keunikan S3 di Outposts, lihat topik berikut.

Topik

Spesifikasi S3 di Outposts

- Ukuran maksimum bucket Outposts adalah 50 TB.
- Jumlah maksimum bucket Outposts adalah 100 per Akun AWS.
- Bucket Outposts hanya dapat diakses menggunakan titik akses dan titik akhir.
- Jumlah maksimum titik akses per bucket Outposts adalah 10.
- Kebijakan titik akses dibatasi sebesar 20 KB ukurannya.
- Pemilik Outpost dapat mengelola akses dalam organisasi Anda AWS Organizations dengan menggunakan AWS Resource Access Manager. Semua akun yang memerlukan akses ke Outpost harus berada dalam organisasi yang sama dengan akun pemilik di AWS Organizations.
- Akun pemilik bucket S3 di Outposts selalu menjadi pemilik semua objek dalam bucket.
- Hanya akun pemilik bucket S3 di Outposts yang dapat melakukan operasi pada bucket.
- Pembatasan ukuran objek sesuai dengan Amazon S3.

- Semua objek yang disimpan dalam S3 di Outposts disimpan di kelas penyimpanan `OUTPOSTS`.
- Secara default, semua objek yang disimpan dalam kelas penyimpanan `OUTPOSTS` disimpan menggunakan enkripsi di sisi server dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Anda juga dapat secara eksplisit memilih untuk menyimpan objek menggunakan enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C).
- Apabila tidak tersedia cukup ruang untuk menyimpan objek di Outpost Anda, API mengembalikan pengecualian kapasitas yang tidak memadai (ICE).

Operasi API yang didukung oleh S3 di Outposts

Untuk mengetahui daftar operasi API yang didukung oleh S3 di Outposts, lihat Operasi API Amazon S3 di Outposts.

AWS CLI Perintah Amazon S3 didukung oleh S3 di Outposts

AWS CLI Perintah Amazon S3 berikut saat ini didukung oleh Amazon S3 di Outposts. Untuk informasi selengkapnya, lihat Perintah yang Tersedia di Referensi AWS CLI Perintah.

- `cp`,`mv`, dan `sync` dalam hal yang sama Outposts ember, atau antara lingkungan lokal dan ember Outposts.
- `ls`
- `presign`
- `rm`

Fitur Amazon S3 yang tidak didukung oleh S3 di Outposts

Fitur Amazon S3 berikut saat ini tidak didukung oleh Amazon S3 di Outposts. Setiap upaya untuk menggunakannya ditolak.

- Permintaan bersyarat
- Daftar kontrol akses (ACLs)
- Berbagi sumber daya lintas asal (CORS)
- Operasi Batch S3
- Laporan Inventaris S3

- Mengubah enkripsi bucket default
- Bucket publik
- Hapus autentikasi multi-faktor (MFA)
- Transisi Siklus Hidup S3 (selain penghapusan objek dan pembatalan pengunggahan multibagian yang tidak lengkap)
- Penahanan Legal Kunci Objek S3
- Retensi Kunci Objek
- Enkripsi sisi server dengan kunci AWS Key Management Service (AWS KMS) (SSE-KMS)
- Kontrol Waktu Replikasi S3 (S3 RTC)
- Metrik CloudWatch permintaan Amazon
- Konfigurasi metrik
- Akselerasi Transfer
- Notifikasi Peristiwa S3
- Bucket Peminta Membayar
- Pilih S3
- AWS Lambda acara
- Pencatatan akses server
- Permintaan HTTP POST
- SOAP
- Akses situs web

Persyaratan jaringan S3 di Outposts

- Untuk merutekan permintaan ke titik akses S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Batas berikut berlaku ke titik akhir untuk S3 di Outposts:
  - Setiap cloud privat virtual (VPC) pada Outposts dapat memiliki satu titik akhir terkait, dan Anda dapat memiliki hingga 100 titik akhir per Outposts.
  - Anda dapat memetakan beberapa titik akses ke titik akhir yang sama.
  - Anda dapat menambahkan titik akhir hanya VPCs dengan blok CIDR di subruang rentang CIDR berikut:
    - 10.0.0.0/8

  - 172.16.0.0/12
  - 192.168.0.0/16
- Anda dapat membuat titik akhir ke Outpost hanya dari yang memiliki blok CIDR VPCs yang tidak tumpang tindih.
- Anda dapat membuat titik akhir hanya dari dalam subnet Outpostsnya.
- Subnet yang Anda gunakan untuk membuat titik akhir harus berisi empat alamat IP untuk digunakan S3 di Outposts.
- Apabila Anda menentukan kumpulan alamat IP milik pelanggan (kumpulan CoIP), kumpulan harus berisi empat alamat IP untuk digunakan S3 di Outposts.
- Anda hanya dapat membuat satu titik akhir per Outpost per VPC.

Memulai dengan Amazon S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API.

Dengan Amazon S3 di Outposts, Anda dapat menggunakan Amazon S3 APIs dan fitur, seperti penyimpanan objek, kebijakan akses, enkripsi, dan penandaan, seperti yang Anda AWS Outposts lakukan di Amazon S3. Untuk informasi tentang S3 di Outposts, lihat Apa itu Amazon S3 di Outposts?

Topik

- Memulai dengan menggunakan AWS Management Console
- Memulai dengan menggunakan AWS CLI dan SDK for Java

Memulai dengan menggunakan AWS Management Console

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Untuk memulai dengan S3 di Outposts menggunakan konsol, lihat topik berikut. Untuk memulai dengan menggunakan AWS CLI atau AWS SDK untuk Java, lihatMemulai dengan menggunakan AWS CLI dan SDK for Java.

Topik

Buat bucket, titik akses, dan titik akhir

Prosedur berikut ini menunjukkan cara membuat bucket pertama Anda di S3 di Outposts. Saat membuat bucket menggunakan konsol, Anda juga membuat titik akses dan titik akhir yang terkait dengan bucket agar Anda dapat segera mulai menyimpan objek di bucket Anda.

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih Buat bucket Outposts.
4. Untuk Nama bucket, masukkan nama yang sesuai dengan Sistem Nama Domain (DNS) untuk bucket Anda.

   Nama bucket harus:

   - Jadilah unik di dalam Akun AWS, Pos Luar, dan tempat Pos Luar berada. Wilayah AWS
   - Memiliki panjang 3-63 karakter.
   - Tidak mengandung karakter huruf besar.
   - Mulai dengan huruf kecil atau angka.

   Setelah membuat bucket, Anda tidak dapat mengubah namanya. Untuk informasi tentang penamaan bucket, lihat Aturan penamaan bucket tujuan umum di Panduan Pengguna Amazon S3.

   > **Important**
   >
   > Hindari menyertakan informasi sensitif, seperti nomor akun, dalam nama bucket. Nama ember terlihat di titik URLs itu ke objek di ember.

5. Untuk Outpost, pilih Outpost tempat Anda ingin bucket berada.
6. Dalam Penentuan Versi Bucket, atur status Penentuan Versi S3 untuk bucket S3 di Outposts Anda ke salah satu opsi berikut:

   - Nonaktifkan (default)-Bucket tetap tidak berversi.
   - Aktifkan-Mengaktifkan Penentuan Versi S3 untuk objek di bucket. Semua objek yang ditambahkan ke bucket menerima ID versi unik.

   Untuk informasi selengkapnya tentang Penentuan Versi S3, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.
7. (Opsional) Tambahkan tag opsional yang ingin Anda kaitkan dengan bucket Outposts. Anda dapat menggunakan tag untuk melacak kriteria bagi proyek individu atau kelompok proyek, atau untuk melabeli bucket Anda menggunakan tag alokasi biaya.

   Secara default, semua objek yang disimpan dalam bucket Outposts disimpan menggunakan enkripsi di sisi server dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Anda juga dapat secara eksplisit memilih untuk menyimpan objek menggunakan enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C). Untuk mengubah jenis enkripsi, Anda harus menggunakan REST API, AWS Command Line Interface (AWS CLI), atau AWS SDKs.
8. Di bagian Pengaturan titik akses Outposts, masukkan nama titik akses.

   Titik akses S3 di Outposts menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di S3 di Outposts. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket Outposts yang dapat Anda gunakan untuk melakukan operasi objek S3. Untuk informasi selengkapnya, lihat Titik Akses.

   Nama titik akses harus unik dalam akun untuk Wilayah dan Pos Terdepan ini, dan mematuhi batasan dan batasan jalur akses.
9. Pilih VPC untuk titik akses Amazon S3 di Outposts ini.

   Jika Anda tidak memiliki VPC, pilih Buat VPC. Untuk informasi selengkapnya, lihat Membuat titik akses yang dibatasi untuk virtual private cloud (VPC) di Panduan Pengguna Amazon S3.

   Cloud privat virtual (VPC) memungkinkan Anda meluncurkan sumber daya AWS ke jaringan virtual yang Anda tentukan. Jaringan virtual ini sangat mirip dengan jaringan tradisional yang

akan Anda operasikan di pusat data Anda sendiri, dengan manfaat dari penggunaan infrastruktur AWS yang dapat diskalakan.

10. (Opsional untuk VPC yang ada) Pilih subnet Titik Akhir untuk titik akhir Anda.

    Subnet adalah serangkaian alamat IP di VPC. Jika Anda tidak memiliki subnet yang Anda inginkan, pilih Buat subnet. Untuk informasi selengkapnya, lihat Jaringan untuk S3 di Outposts.

11. (Opsional untuk VPC yang ada) Pilih Grup keamanan Titik Akhir untuk titik akhir Anda.

    Grup keamanan bertindak sebagai firewall virtual untuk mengendalikan lalu lintas masuk dan keluar.

12. (Opsional untuk VPC yang ada) Pilih jenis akses Titik akhir:

    - Privat–Untuk digunakan dengan VPC.
    - IP milik pelanggan–Untuk digunakan dengan kumpulan alamat IP milik pelanggan (kumpulan CoIP) dari dalam jaringan on-premise Anda.

13. (Opsional) Tentukan kebijakan titik akses Outpost. Konsol secara otomatis menampilkan Amazon Resource Name (ARN) untuk titik akses, yang dapat Anda gunakan dalam kebijakan.

14. Pilih Buat bucket Outposts.

> **Note**
>
> Diperlukan waktu hingga 5 menit untuk membuat titik akhir Outposts Anda dan bucket Anda siap digunakan. Untuk mengonfigurasi pengaturan bucket tambahan, pilih Lihat detail.

Langkah selanjutnya

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Setelah membuat bucket, titik akses, dan titik akhir S3 di Outposts, Anda dapat menggunakan AWS CLI atau SDK for Java untuk mengunggah objek ke bucket. Untuk informasi selengkapnya, lihat Unggah objek ke ember S3 di Outposts.

Memulai dengan menggunakan AWS CLI dan SDK for Java

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts OUTPOSTS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, lihat Apa itu Amazon S3 di Outposts?

Untuk memulai S3 di Outposts, Anda harus membuat bucket, titik akses, dan titik akhir. Kemudian, Anda dapat mengunggah objek ke bucket Anda. Contoh berikut menunjukkan kepada Anda bagaimana memulai dengan S3 di Outposts dengan menggunakan AWS CLI dan SDK for Java. Untuk memulai menggunakan konsol, lihat Memulai dengan menggunakan AWS Management Console.

Topik

- Langkah 1: Buat bucket
- Langkah 2: Buat titik akses
- Langkah 3: Buat titik akhir
- Langkah 4: Unggah objek ke bucket S3 di Outposts

Langkah 1: Buat bucket

Contoh berikut AWS CLI dan SDK for Java menunjukkan cara membuat bucket S3 di Outposts.

AWS CLI

Example

Contoh berikut membuat bucket S3 di Outposts (`s3-outposts:CreateBucket`) menggunakan AWS CLI. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control create-bucket --bucket example-outposts-bucket --outpost-
id op-01ac5d28a6a232904
```

SDK for Java

Example

Untuk contoh cara membuat bucket Outposts S3 dengan AWS SDK for Java CreateOutpostsBucket, lihat .java di SDK for Java 2.x Code AWS Examples.

Langkah 2: Buat titik akses

Untuk mengakses bucket Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasi titik akses. Ini contoh bagaimana Anda cara membuat titik akses dengan menggunakan AWS CLI dan SDK for Java.

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

AWS CLI

Example

AWS CLI Contoh berikut membuat titik akses untuk bucket Outposts. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control create-access-point --account-id 123456789012
 --name example-outposts-access-point --bucket "arn:aws:s3-
```

```
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket" --vpc-configuration VpcId=example-vpc-12345
```

SDK for Java

Example

Untuk contoh cara membuat titik akses untuk bucket S3 Outposts dengan AWS SDK for Java CreateOutpostsAccessPoint, lihat AWS .java di SDK for Java 2.x Code Examples.

Langkah 3: Buat titik akhir

Untuk merutekan permintaan ke titik akses Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Untuk membuat titik akhir, Anda akan memerlukan koneksi aktif dengan tautan layanan ke wilayah asal Outposts Anda. Setiap Cloud Privat Virtual (VPC) pada Outposts Anda dapat memiliki satu titik akhir terkait. Untuk informasi selengkapnya tentang kuota titik akhir, lihat Persyaratan jaringan S3 di Outposts. Anda harus membuat titik akhir agar dapat mengakses bucket Outposts Anda dan melakukan operasi objek. Untuk informasi selengkapnya, lihat Titik akhir.

Contoh-contoh ini menunjukkan cara membuat endpoint dengan menggunakan AWS CLI dan SDK for Java. Untuk informasi selengkapnya tentang izin yang diperlukan untuk membuat dan mengelola titik akhir, lihat Izin untuk titik akhir S3 di Outposts.

AWS CLI

Example

AWS CLI Contoh berikut membuat endpoint untuk Outpost dengan menggunakan jenis akses sumber daya VPC. VPC berasal dari subnet. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1
```

AWS CLI Contoh berikut membuat endpoint untuk Outpost dengan menggunakan jenis akses pool alamat IP (CoIP pool) milik pelanggan. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1 --access-type CustomerOwnedIp --
customer-owned-ipv4-pool ipv4pool-coip-12345678901234567
```

SDK for Java

Example

Untuk contoh cara membuat endpoint untuk Outpost S3 dengan SDK for Java, CreateOutpostsEndPointlihat AWS .java di SDK for Java 2.x Code Examples.AWS

Langkah 4: Unggah objek ke bucket S3 di Outposts

Untuk mengunggah objek, lihatUnggah objek ke ember S3 di Outposts.

Jaringan untuk S3 di Outposts

Anda dapat menggunakan S3 di Outposts untuk menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data, dan residensi data. Bagian ini menjelaskan persyaratan jaringan untuk mengakses S3 di Outposts.

Topik

Memilih jenis akses jaringan Anda

Anda dapat mengakses S3 di Outposts dari dalam VPC atau dari jaringan lokal Anda. Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir. Koneksi ini membuat lalu lintas antara VPC Anda dan S3 Anda di bucket Outposts di dalam jaringan. AWS Saat Anda membuat titik akhir, Anda harus menentukan jenis akses titik akhir sebagai (untuk perutean VPC) atau `Private` `CustomerOwnedIp` (untuk kumpulan alamat IP milik pelanggan [CoIP pool]).

- `Private` (untuk perutean VPC)-Jika Anda tidak menentukan jenis akses, S3 di Outposts menggunakan secara default `Private`. Dengan tipe `Private` akses, instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan sumber daya di Outposts Anda. Anda dapat bekerja dengan S3 di Outposts dari dalam VPC. Jenis titik akhir ini dapat diakses dari jaringan lokal Anda melalui perutean VPC langsung. Untuk informasi selengkapnya, lihat Tabel rute gateway lokal di Panduan Pengguna AWS Outposts.
- `CustomerOwnedIp`(untuk kumpulan CoIP)-Jika Anda tidak default ke jenis `Private` akses dan memilih`CustomerOwnedIp`, Anda harus menentukan rentang alamat IP. Anda dapat menggunakan jenis akses ini untuk bekerja dengan S3 di Outposts baik dari jaringan lokal maupun dalam VPC. Saat mengakses S3 di Outposts dalam VPC, lalu lintas Anda terbatas pada bandwidth gateway lokal.

Mengakses S3 Anda di bucket dan objek Outposts

Untuk mengakses S3 Anda di bucket Outposts dan objek, Anda harus memiliki berikut ini:

- Titik akses untuk VPC.
- Titik Akhir untuk VPC yang sama.
- Koneksi aktif antara Outpost Anda dan Anda Wilayah AWS. Untuk informasi selengkapnya tentang cara menghubungkan Pos Luar Anda ke Wilayah, lihat Konektivitas pos terdepan ke AWS Wilayah di Panduan Pengguna AWS Outposts.

Untuk informasi lebih lanjut tentang mengakses bucket dan objek di S3 di Outposts, lihat dan. Bekerja dengan S3 di bucket Outposts Bekerja dengan S3 di objek Outposts

Antarmuka jaringan elastis lintas akun

Titik akhir S3 di Outposts diberi nama resource dengan Amazon Resource Names (). ARNs Saat titik akhir ini dibuat, siapkan beberapa AWS Outposts antarmuka jaringan elastis lintas akun. Antarmuka jaringan elastis lintas akun S3 di Outposts seperti antarmuka jaringan lainnya dengan satu pengecualian: S3 di Outposts mengaitkan antarmuka jaringan elastis lintas akun ke instans Amazon. EC2

S3 di Sistem Nama Domain (DNS) Outposts menyeimbangkan beban permintaan Anda melalui antarmuka jaringan elastisitas lintas akun. S3 on Outposts membuat cross-account elastic network interface di akun AWS Anda yang terlihat dari panel Network interface dari konsol Amazon. EC2

Untuk titik akhir yang menggunakan tipe akses pool CoIP, S3 on Outposts mengalokasikan dan mengaitkan alamat IP dengan antarmuka jaringan elastis lintas akun dari pool CoIP yang dikonfigurasi.

Bekerja dengan S3 di bucket Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Anda berkomunikasi dengan bucket Outpost Anda dengan menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Untuk mengakses S3 Anda di bucket dan objek Outposts, Anda harus memiliki titik akses untuk VPC dan titik akhir untuk VPC yang sama. Untuk informasi selengkapnya, lihat Jaringan untuk S3 di Outposts.

Bucket

Di S3 di Outposts, nama bucket unik untuk Outpost dan memerlukan kode untuk Wilayah Wilayah AWS tempat Pos Luar Akun AWS , ID, ID Pos Luar, dan nama bucket untuk mengidentifikasinya.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/bucket/bucket-name
```

Untuk informasi selengkapnya, lihat Sumber daya ARNs untuk S3 di Outposts.

Titik Akses

Amazon S3 di Outposts mendukung titik akses khusus virtual private cloud (VPC) sebagai satu-satunya cara untuk mengakses bucket Outposts.

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

Contoh berikut menunjukkan format ARN yang Anda gunakan untuk S3 di titik akses Outposts. Titik akses ARN mencakup Wilayah AWS kode untuk Wilayah Pos Luar, Akun AWS ID, ID Pos Luar, dan nama titik akses.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Titik akhir

Untuk merutekan permintaan ke titik akses S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Dengan S3 pada titik akhir Outposts, Anda dapat menghubungkan VPC Anda secara pribadi ke bucket Outpost Anda. S3 pada titik akhir Outposts adalah pengidentifikasi sumber daya seragam virtual URIs () dari titik masuk ke bucket S3 Anda di Outposts. Mereka merupakan komponen VPC skala horizontal, redundan, dan sangat tersedia.

Setiap cloud privat virtual (VPC) pada Outpost Anda dapat memiliki satu titik akhir terkait, dan Anda dapat memiliki hingga 100 titik akhir per Outpost. Anda harus membuat titik akhir ini agar dapat mengakses bucket Outpost Anda dan melakukan operasi objek. Membuat titik akhir ini juga memungkinkan model dan perilaku API menjadi sama dengan memungkinkan operasi yang sama untuk bekerja di S3 dan S3 di Outposts.

Operasi API di S3 di Outposts

Untuk mengelola operasi API bucket Outpost, S3 di Outposts menjadi host titik akhir terpisah yang berbeda dari titik akhir Amazon S3. Titik akhir ini adalah `s3-outposts.region.amazonaws.com`.

Untuk menggunakan operasi Amazon S3 API, Anda harus menandatangani bucket dan objek menggunakan format ARN yang benar. Anda harus ARNs meneruskan operasi API sehingga Amazon S3 dapat menentukan apakah permintaan tersebut untuk Amazon S3 `s3-control.region.amazonaws.com` () atau untuk S3 di Outposts (). `s3-outposts.region.amazonaws.com` Berdasarkan format ARN, S3 kemudian dapat menandatangani dan merutekan permintaan dengan tepat.

Setiap kali permintaan dikirim ke bidang kontrol Amazon S3, SDK mengekstraksi komponen dari ARN dan menyertakan header tambahan`x-amz-outpost-id`, dengan nilai yang *outpost-id* diambil dari ARN. Nama layanan dari ARN digunakan untuk menandai permintaan sebelum diarahkan ke S3 di titik akhir Outposts. Perilaku ini berlaku untuk semua operasi API yang ditangani oleh `s3control` klien.

Tabel berikut mencantumkan operasi API yang diperluas untuk Amazon S3 di Outposts dan perubahannya yang relatif terhadap Amazon S3.

API	S3 pada nilai parameter Outposts	
`CreateBucket`	Nama bucket sebagai ARN, Outpost	
`ListRegionalBuckets`	outpost-id	
`DeleteBucket`	nama bucket sebagai ARN	
`DeleteBucketLifecycleConfiguration`	nama bucket sebagai ARN	
`GetBucketLifecycleConfiguration`	nama bucket sebagai ARN	
`PutBucketLifecycleConfiguration`	nama bucket sebagai ARN	
`GetBucketPolicy`	nama bucket sebagai ARN	
`PutBucketPolicy`	nama bucket sebagai ARN	
`DeleteBucketPolicy`	nama bucket sebagai ARN	
`GetBucketTagging`	nama bucket sebagai ARN	
`PutBucketTagging`	nama bucket sebagai ARN	
`DeleteBucketTagging`	nama bucket sebagai ARN	
`CreateAccessPoint`	Nama titik akses sebagai ARN	
`DeleteAccessPoint`	Nama titik akses sebagai ARN	
`GetAccessPoint`	Nama titik akses sebagai ARN	
`GetAccessPoint`	Nama titik akses sebagai ARN	

API	S3 pada nilai parameter Outposts	
`ListAccessPoints`	Nama titik akses sebagai ARN	
`PutAccessPointPolicy`	Nama titik akses sebagai ARN	
`GetAccessPointPolicy`	Nama titik akses sebagai ARN	
`DeleteAccessPointPolicy`	Nama titik akses sebagai ARN	

Membuat dan mengelola S3 di bucket Outposts

Untuk informasi selengkapnya tentang membuat dan mengelola S3 di bucket Outposts, lihat topik berikut.

Membuat bucket S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts `OUTPOSTS` (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, lihat Apa itu Amazon S3 di Outposts?

> **Note**
>
> Akun AWS Yang menciptakan ember memilikinya dan merupakan satu-satunya yang dapat melakukan tindakan untuk itu. Bucket memiliki properti konfigurasi, seperti Outpost, tag, enkripsi default, dan pengaturan titik akses. Pengaturan titik akses termasuk cloud privat virtual (VPC), kebijakan titik akses untuk mengakses objek di bucket, dan metadata lainnya. Untuk informasi selengkapnya, lihat Spesifikasi S3 di Outposts.

Jika Anda ingin membuat bucket yang digunakan AWS PrivateLink untuk menyediakan akses manajemen bucket dan endpoint melalui titik akhir VPC antarmuka di virtual private cloud (VPC) Anda, lihat AWS PrivateLink S3 di Outposts.

Contoh berikut menunjukkan cara membuat bucket S3 di Outposts dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih Buat bucket Outposts.
4. Untuk Nama bucket, masukkan nama yang sesuai dengan Sistem Nama Domain (DNS) untuk bucket Anda.

   Nama bucket harus:

   - Jadilah unik di dalam Akun AWS, Pos Luar, dan tempat Pos Luar berada. Wilayah AWS
   - Memiliki panjang 3-63 karakter.
   - Tidak mengandung karakter huruf besar.
   - Mulai dengan huruf kecil atau angka.

   Setelah membuat bucket, Anda tidak dapat mengubah namanya. Untuk informasi tentang penamaan bucket, lihat Aturan penamaan bucket tujuan umum di Panduan Pengguna Amazon S3.

   > **Important**
   >
   > Hindari menyertakan informasi sensitif, seperti nomor akun, dalam nama bucket. Nama ember terlihat di titik URLs itu ke objek di ember.

5. Untuk Outpost, pilih Outpost tempat Anda ingin bucket berada.
6. Dalam Penentuan Versi Bucket, atur status Penentuan Versi S3 untuk bucket S3 di Outposts Anda ke salah satu opsi berikut:

   - Nonaktifkan (default)-Bucket tetap tidak berversi.

- Aktifkan-Mengaktifkan Penentuan Versi S3 untuk objek di bucket. Semua objek yang ditambahkan ke bucket menerima ID versi unik.

Untuk informasi selengkapnya tentang Penentuan Versi S3, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.

7. (Opsional) Tambahkan tag opsional yang ingin Anda kaitkan dengan bucket Outposts. Anda dapat menggunakan tag untuk melacak kriteria bagi proyek individu atau kelompok proyek, atau untuk melabeli bucket Anda menggunakan tag alokasi biaya.

   Secara default, semua objek yang disimpan dalam bucket Outposts disimpan menggunakan enkripsi di sisi server dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Anda juga dapat secara eksplisit memilih untuk menyimpan objek menggunakan enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C). Untuk mengubah jenis enkripsi, Anda harus menggunakan REST API, AWS Command Line Interface (AWS CLI), atau AWS SDKs.

8. Di bagian Pengaturan titik akses Outposts, masukkan nama titik akses.

   Titik akses S3 di Outposts menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di S3 di Outposts. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket Outposts yang dapat Anda gunakan untuk melakukan operasi objek S3. Untuk informasi selengkapnya, lihat Titik Akses.

   Nama titik akses harus unik dalam akun untuk Wilayah dan Pos Terdepan ini, dan mematuhi batasan dan batasan jalur akses.

9. Pilih VPC untuk titik akses Amazon S3 di Outposts ini.

   Jika Anda tidak memiliki VPC, pilih Buat VPC. Untuk informasi selengkapnya, lihat Membuat titik akses yang dibatasi untuk virtual private cloud (VPC) di Panduan Pengguna Amazon S3.

   Cloud privat virtual (VPC) memungkinkan Anda meluncurkan sumber daya AWS ke jaringan virtual yang Anda tentukan. Jaringan virtual ini sangat mirip dengan jaringan tradisional yang akan Anda operasikan di pusat data Anda sendiri, dengan manfaat dari penggunaan infrastruktur AWS yang dapat diskalakan.

10. (Opsional untuk VPC yang ada) Pilih subnet Titik Akhir untuk titik akhir Anda.

    Subnet adalah serangkaian alamat IP di VPC. Jika Anda tidak memiliki subnet yang Anda inginkan, pilih Buat subnet. Untuk informasi selengkapnya, lihat Jaringan untuk S3 di Outposts.

11. (Opsional untuk VPC yang ada) Pilih Grup keamanan Titik Akhir untuk titik akhir Anda.

    Grup keamanan bertindak sebagai firewall virtual untuk mengendalikan lalu lintas masuk dan keluar.

12. (Opsional untuk VPC yang ada) Pilih jenis akses Titik akhir:

    - Privat–Untuk digunakan dengan VPC.
    - IP milik pelanggan–Untuk digunakan dengan kumpulan alamat IP milik pelanggan (kumpulan CoIP) dari dalam jaringan on-premise Anda.

13. (Opsional) Tentukan kebijakan titik akses Outpost. Konsol secara otomatis menampilkan Amazon Resource Name (ARN) untuk titik akses, yang dapat Anda gunakan dalam kebijakan.

14. Pilih Buat bucket Outposts.

> **Note**
>
> Diperlukan waktu hingga 5 menit untuk membuat titik akhir Outposts Anda dan bucket Anda siap digunakan. Untuk mengonfigurasi pengaturan bucket tambahan, pilih Lihat detail.

Menggunakan AWS CLI

Example

Contoh berikut membuat bucket S3 di Outposts (`s3-outposts:CreateBucket`) menggunakan AWS CLI. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control create-bucket --bucket example-outposts-bucket --outpost-
id op-01ac5d28a6a232904
```

Menggunakan AWS SDK for Java

Example

Untuk contoh cara membuat bucket Outposts S3 dengan AWS SDK for Java CreateOutpostsBucket, lihat .java di SDK for Java 2.x Code AWS Examples.

Menambahkan tag untuk bucket S3 pada Outposts

Anda dapat menambahkan tag untuk bucket Amazon S3 di Outposts untuk melacak biaya penyimpanan dan kriteria lain untuk proyek individu atau grup proyek.

> **Note**
>
> Akun AWS Yang membuat ember memilikinya dan merupakan satu-satunya yang dapat mengubah tagnya.

Menggunakan konsol S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang tag ingin Anda edit.
4. Pilih tab Properti.
5. Di bawah Tag, pilih Edit.
6. Pilih Tambahkan tag baru, dan masukkan Kunci dan Nilai opsional.

   Tambahkan tag apa pun yang ingin Anda kaitkan dengan bucket Outposts untuk melacak kriteria lain untuk proyek individu atau grup proyek.
7. Pilih Simpan perubahan.

Menggunakan AWS CLI

AWS CLI Contoh berikut menerapkan konfigurasi penandaan ke bucket S3 pada Outposts dengan menggunakan dokumen JSON di folder saat ini yang menentukan tag (). *`tagging.json`* Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

```
aws s3control put-bucket-tagging --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --tagging file://tagging.json
```

```
tagging.json

{
   "TagSet": [
     {
       "Key": "organization",
       "Value": "marketing"
     }
   ]
}
```

AWS CLI Contoh berikut menerapkan konfigurasi penandaan ke bucket S3 pada Outposts langsung dari baris perintah.

```
aws s3control put-bucket-tagging --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --tagging 'TagSet=[{Key=organization,Value=marketing}]'
```

Untuk informasi selengkapnya tentang perintah ini, lihat put-bucket-taggingdi AWS CLI Referensi.

Mengelola akses ke bucket Amazon S3 di Outposts menggunakan kebijakan bucket

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya. Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB. Untuk informasi selengkapnya, lihat Kebijakan bucket.

Anda dapat memperbarui kebijakan bucket untuk mengelola akses ke bucket Amazon S3 di Outposts. Untuk informasi selengkapnya, lihat topik berikut.

Topik

- Menambahkan atau mengedit kebijakan bucket untuk bucket Amazon S3 di Outposts
- Melihat kebijakan bucket untuk bucket Amazon S3 di Outposts Anda
- Menghapus kebijakan bucket untuk bucket Amazon S3 di Outposts Anda
- Contoh kebijakan bucket

Menambahkan atau mengedit kebijakan bucket untuk bucket Amazon S3 di Outposts

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya. Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB. Untuk informasi selengkapnya, lihat Kebijakan bucket.

Topik berikut menunjukkan cara memperbarui kebijakan bucket Amazon S3 pada Outposts dengan menggunakan AWS Management Console, AWS Command Line Interface ()AWS CLI, atau. AWS SDK untuk Java

Menggunakan konsol S3

Untuk membuat atau mengedit kebijakan bucket

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang kebijakan bucketnya ingin Anda edit.
4. Pilih tab Izin.
5. Di bagian kebijakan bucket Outposts, untuk membuat atau mengedit kebijakan baru, pilih Edit.

   Anda sekarang dapat menambah atau mengedit kebijakan bucket S3 di Outposts. Untuk informasi selengkapnya, lihat Mengatur IAM dengan S3 di Outposts.

Menggunakan AWS CLI

AWS CLI Contoh berikut menempatkan kebijakan pada bucket Outposts.

1. Simpan kebijakan bucket berikut ke file JSON. Dalam contoh ini, file dinamai `policy1.json`. Ganti *`user input placeholders`* dengan informasi Anda sendiri.

   JSON

```
{
```

```
   "Version":"2012-10-17",
   "Id":"testBucketPolicy",
   "Statement":[
      {
         "Sid":"st1",
         "Effect":"Allow",
         "Principal":{
            "AWS":"arn:aws:iam::123456789012:root"
         },
         "Action":[
            "s3-outposts:GetObject",
            "s3-outposts:PutObject",
            "s3-outposts:DeleteObject",
            "s3-outposts:ListBucket"
         ],
         "Resource":"arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/amzn-s3-
demo-bucket"
      }
   ]
}
```

2. Kirim file JSON sebagai bagian dari perintah CLI `put-bucket-policy`. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control put-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --policy file://policy1.json
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut menempatkan kebijakan pada bucket Outposts.

```
import com.amazonaws.services.s3control.model.*;

public void putBucketPolicy(String bucketArn) {

    String policy = "{\"Version\":\"2012-10-17\",\"Id\":\"testBucketPolicy\",
\"Statement\":[{\"Sid\":\"st1\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"" +
 AccountId+ "\"},\"Action\":\"s3-outposts:*\",\"Resource\":\"" + bucketArn + "\"}]}";

    PutBucketPolicyRequest reqPutBucketPolicy = new PutBucketPolicyRequest()
```

```
            .withAccountId(AccountId)
            .withBucket(bucketArn)
            .withPolicy(policy);

    PutBucketPolicyResult respPutBucketPolicy =
 s3ControlClient.putBucketPolicy(reqPutBucketPolicy);
    System.out.printf("PutBucketPolicy Response: %s%n",
 respPutBucketPolicy.toString());

}
```

Melihat kebijakan bucket untuk bucket Amazon S3 di Outposts Anda

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya. Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB. Untuk informasi selengkapnya, lihat Kebijakan bucket.

Topik berikut menunjukkan cara melihat kebijakan bucket Amazon S3 di Outposts dengan menggunakan AWS Management Console, AWS Command Line Interface ()AWS CLI, atau. AWS SDK untuk Java

Menggunakan konsol S3

Untuk membuat atau mengedit kebijakan bucket

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang izinnya ingin Anda edit.
4. Pilih tab Izin.
5. Di bagian kebijakan bucket Outposts, Anda dapat meninjau kebijakan bucket yang ada. Untuk informasi selengkapnya, lihat Mengatur IAM dengan S3 di Outposts.

Menggunakan AWS CLI

AWS CLI Contoh berikut mendapatkan kebijakan untuk bucket Outposts. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control get-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mendapatkan kebijakan untuk bucket Outposts.

```
import com.amazonaws.services.s3control.model.*;

public void getBucketPolicy(String bucketArn) {

    GetBucketPolicyRequest reqGetBucketPolicy = new GetBucketPolicyRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    GetBucketPolicyResult respGetBucketPolicy =
 s3ControlClient.getBucketPolicy(reqGetBucketPolicy);
    System.out.printf("GetBucketPolicy Response: %s%n",
 respGetBucketPolicy.toString());

}
```

Menghapus kebijakan bucket untuk bucket Amazon S3 di Outposts Anda

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya. Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB. Untuk informasi selengkapnya, lihat Kebijakan bucket.

Topik berikut menunjukkan cara melihat kebijakan bucket Amazon S3 di Outposts dengan menggunakan or (). AWS Management Console AWS Command Line Interface AWS CLI

Menggunakan konsol S3

Untuk menghapus kebijakan bucket

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang izinnya ingin Anda edit.

4. Pilih tab Izin.
5. Di bagian kebijakan bucket Outposts, pilih Hapus.
6. Konfirmasi penghapusan.

Menggunakan AWS CLI

Contoh berikut menghapus kebijakan bucket untuk bucket S3 di Outposts (`s3-outposts:DeleteBucket`) menggunakan AWS CLI. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control delete-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket
```

Contoh kebijakan bucket

Dengan kebijakan bucket S3 on Outposts, Anda dapat mengamankan akses ke objek di bucket S3 di Outposts, sehingga hanya pengguna dengan izin yang sesuai yang dapat mengaksesnya. Anda bahkan dapat mencegah pengguna yang diautentikasi tanpa izin yang sesuai untuk mengakses sumber daya S3 Anda di Outposts.

Bagian ini menyajikan contoh kasus penggunaan umum untuk kebijakan bucket S3 pada Outposts. Untuk menguji kebijakan ini, ganti *`user input placeholders`* dengan informasi Anda sendiri (seperti nama bucket Anda).

Untuk memberikan atau menolak izin ke sekumpulan objek, Anda dapat menggunakan karakter wildcard (*) di Amazon Resource Names (ARNs) dan nilai lainnya. Misalnya, Anda dapat mengendalikan akses ke kelompok objek yang dimulai dengan prefiks umum atau diakhiri dengan ekstensi tertentu, seperti `.html`.

Untuk informasi selengkapnya tentang bahasa kebijakan AWS Identity and Access Management (IAM), lihatMengatur IAM dengan S3 di Outposts.

> **Note**
>
> Saat menguji s3outpostsizin menggunakan konsol Amazon S3, Anda harus memberikan izin tambahan yang diperlukan konsol, `s3outposts:createendpoint` seperti,, dan `s3outposts:listendpoints` sebagainya.

Sumber daya tambahan untuk membuat kebijakan bucket

- Untuk daftar tindakan kebijakan IAM, sumber daya, dan kunci kondisi yang dapat Anda gunakan saat membuat kebijakan bucket S3 di Outposts, lihat Kunci tindakan, sumber daya, dan kondisi untuk Amazon S3 di Outposts.
- Untuk panduan cara membuat kebijakan S3 tentang Outposts Anda, lihat. Menambahkan atau mengedit kebijakan bucket untuk bucket Amazon S3 di Outposts

Topik

- Mengelola akses ke bucket Amazon S3 di Outposts berdasarkan alamat IP tertentu

Mengelola akses ke bucket Amazon S3 di Outposts berdasarkan alamat IP tertentu

Kebijakan bucket adalah kebijakan berbasis sumber daya AWS Identity and Access Management (IAM) yang dapat Anda gunakan untuk memberikan izin akses ke bucket dan objek di dalamnya. Hanya pemilik bucket yang dapat mengaitkan kebijakan dengan bucket. Izin yang dipasang pada bucket berlaku untuk semua objek di bucket yang dimiliki oleh pemilik bucket. Kebijakan bucket dibatasi hingga ukuran 20 KB. Untuk informasi selengkapnya, lihat Kebijakan bucket.

Membatasi akses ke suatu Wilayah tertentu

Contoh berikut menyangkal semua pengguna melakukan operasi S3 pada Outposts pada objek dalam bucket yang ditentukan kecuali permintaan tersebut berasal dari rentang alamat IP yang ditentukan.

> **Note**
>
> Saat membatasi akses ke alamat IP tertentu, pastikan Anda juga menentukan titik akhir VPC, alamat IP sumber VPC, atau alamat IP eksternal yang dapat mengakses bucket S3 on Outposts. Jika tidak, Anda mungkin kehilangan akses ke bucket jika kebijakan Anda menolak semua pengguna untuk melakukan s3outpostsoperasi apa pun pada objek di bucket S3 di Outposts tanpa izin yang tepat.

`Condition`Pernyataan kebijakan ini mengidentifikasi *`192.0.2.0/24`* sebagai rentang alamat IP versi 4 (IPv4) IP yang diizinkan.

`Condition`Blok menggunakan `NotIpAddress` kondisi dan kunci `aws:SourceIp` kondisi, yang merupakan kunci kondisi AWS lebar. Kunci kondisi `aws:SourceIp` hanya dapat digunakan untuk rentang alamat IP publik. Untuk informasi selengkapnya tentang kunci kondisi ini, lihat Tindakan, sumber daya, dan kunci kondisi untuk S3 di Outposts. `aws:SourceIp` IPv4 Nilai-nilai menggunakan notasi CIDR standar. Untuk informasi selengkapnya, lihat referensi elemen kebijakan IAM JSON di Panduan Pengguna IAM.

> **Warning**
>
> Sebelum menggunakan kebijakan S3 on Outposts ini, ganti *192.0.2.0/24* rentang alamat IP dalam contoh ini dengan nilai yang sesuai untuk kasus penggunaan Anda. Jika tidak, Anda akan kehilangan kemampuan untuk mengakses bucket Anda.

```
{
    "Version": "2012-10-17",
    "Id": "S3OutpostsPolicyId1",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3-outposts:*",
            "Resource": [
                "arn:aws:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
accesspoint/EXAMPLE-ACCESS-POINT-NAME",
                "arn:aws:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
bucket/amzn-s3-demo-bucket"
            ],
            "Condition": {
                "NotIpAddress": {
                    "aws:SourceIp": "192.0.2.0/24"
                }
            }
        }
    ]
}
```

Izinkan keduanya IPv4 dan IPv6 alamat

Saat Anda mulai menggunakan IPv6 alamat, sebaiknya Anda memperbarui semua kebijakan organisasi dengan rentang IPv6 alamat selain IPv4 rentang yang ada. Melakukan hal ini akan membantu memastikan bahwa kebijakan terus berfungsi saat Anda melakukan transisi IPv6.

Kebijakan bucket contoh S3 on Outposts berikut menunjukkan cara IPv4 mencampur IPv6 dan rentang alamat untuk mencakup semua alamat IP valid organisasi Anda. Kebijakan contoh memungkinkan akses ke alamat IP contoh *192.0.2.1*dan *2001:DB8:1234:5678::1*dan menolak akses ke alamat *203.0.113.1*dan *2001:DB8:1234:5678:ABCD::1*.

Kunci kondisi `aws:SourceIp` hanya dapat digunakan untuk rentang alamat IP publik. IPv6 Nilai untuk `aws:SourceIp` harus dalam format CIDR standar. Untuk IPv6, kami mendukung penggunaan `::` untuk mewakili rentang 0 (misalnya,`2001:DB8:1234:5678::/64`). Untuk informasi selengkapnya, lihat operator kondisi alamat IP di Panduan Pengguna IAM.

> **Warning**
>
> Ganti rentang alamat IP dalam contoh ini dengan nilai yang sesuai untuk kasus penggunaan Anda sebelum menggunakan kebijakan S3 on Outposts ini. Jika tidak, Anda mungkin kehilangan kemampuan untuk mengakses bucket Anda.

JSON

```
{
    "Id": "S3OutpostsPolicyId2",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowIPmix",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": [
                "s3-outposts:GetObject",
                "s3-outposts:PutObject",
                "s3-outposts:ListBucket"
            ],
```

```
            "Resource": [
                "arn:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
bucket/amzn-s3-demo-bucket",
                "arn:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
bucket/amzn-s3-demo-bucket/*"
            ],
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": [
                        "192.0.2.0/24",
                        "2001:DB8:1234:5678::/64"
                    ]
                },
                "NotIpAddress": {
                    "aws:SourceIp": [
                        "203.0.113.0/24",
                        "2001:DB8:1234:5678:ABCD::/80"
                    ]
                }
            }
        }
    ]
}
```

Mencantumkan bucket Amazon S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts OUTPOSTS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Untuk informasi selengkapnya tentang bekerja dengan bucket dalam S3 di Outposts, lihat Bekerja dengan S3 di bucket Outposts.

Contoh berikut menunjukkan kepada Anda cara mengembalikan daftar S3 Anda di bucket Outposts dengan menggunakan,, AWS Management Console dan AWS CLI. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Dalam bucket Outposts, tinjau daftar bucket S3 di Outposts.

Menggunakan AWS CLI

AWS CLI Contoh berikut mendapat daftar ember di Outpost. Untuk menggunakan perintah ini, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat list-regional-bucketsdi AWS CLI Referensi.

```
aws s3control list-regional-buckets --account-id 123456789012 --outpost-
id op-01ac5d28a6a232904
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mendapatkan daftar bucket di Outposts. Untuk informasi selengkapnya, lihat ListRegionalBuckets di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void listRegionalBuckets() {

    ListRegionalBucketsRequest reqListBuckets = new ListRegionalBucketsRequest()
            .withAccountId(AccountId)
            .withOutpostId(OutpostId);

    ListRegionalBucketsResult respListBuckets =
 s3ControlClient.listRegionalBuckets(reqListBuckets);
    System.out.printf("ListRegionalBuckets Response: %s%n",
 respListBuckets.toString());

}
```

Mendapatkan bucket S3 di Outposts dengan menggunakan dan AWS CLI SDK for Java

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts OUTPOSTS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Contoh berikut menunjukkan cara mendapatkan bucket S3 di Outposts dengan menggunakan AWS CLI dan. AWS SDK untuk Java

> **Note**
>
> Saat Anda bekerja dengan Amazon S3 di Outposts melalui AWS CLI atau AWS SDKs, Anda memberikan titik akses ARN untuk Outpost sebagai pengganti nama bucket. Titik akses ARN mengikuti bentuk berikut, dengan *region* adalah kode Wilayah AWS untuk Wilayah tempat Outpost berada:
> `arn:aws:s3-outposts:`*region*`:`*123456789012*`:outpost/`*op-01ac5d28a6a232904*`/accesspoint/`*example-outposts-access-point*
> Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Menggunakan AWS CLI

Contoh S3 pada Outposts berikut mendapatkan bucket menggunakan AWS CLI. Untuk menggunakan perintah ini, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat get-bucket dalam Referensi AWS CLI .

```
aws s3control get-bucket --account-id 123456789012 --bucket "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket"
```

Menggunakan AWS SDK for Java

Contoh S3 pada Outposts berikut mendapatkan bucket menggunakan SDK for Java. Untuk informasi selengkapnya, lihat GetBucket di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void getBucket(String bucketArn) {

    GetBucketRequest reqGetBucket = new GetBucketRequest()
            .withBucket(bucketArn)
            .withAccountId(AccountId);

    GetBucketResult respGetBucket = s3ControlClient.getBucket(reqGetBucket);
    System.out.printf("GetBucket Response: %s%n", respGetBucket.toString());

}
```

Menghapus bucket Amazon S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Untuk informasi selengkapnya tentang bekerja dengan bucket dalam S3 di Outposts, lihat Bekerja dengan S3 di bucket Outposts.

Akun AWS Yang membuat ember memilikinya dan merupakan satu-satunya yang dapat menghapusnya.

> **Note**
>
> - Bucket Outposts harus kosong sebelum dapat dihapus.
>
> Konsol Amazon S3 tidak mendukung tindakan objek S3 di Outposts. Untuk menghapus objek di bucket S3 di Outposts, Anda harus menggunakan REST API AWS CLI,, atau. AWS SDKs
> - Sebelum Anda dapat menghapus bucket Outposts, Anda harus menghapus titik akses Outposts untuk bucket tersebut. Untuk informasi selengkapnya, lihat Menghapus titik akses.
> - Anda tidak dapat memulihkan bucket setelah dihapus.

Contoh berikut menunjukkan cara menghapus bucket S3 di Outposts dengan menggunakan AWS Management Console AWS Command Line Interface dan AWS CLI().

Menggunakan konsol S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket yang ingin Anda hapus, dan pilih Hapus.
4. Konfirmasi penghapusan.

Menggunakan AWS CLI

Contoh berikut menghapus bucket S3 di Outposts (`s3-outposts:DeleteBucket`) menggunakan AWS CLI. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control delete-bucket --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Bekerja dengan titik akses Amazon S3 di Outposts

Untuk mengakses bucket Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasi titik akses.

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

> **Note**
>
> Bucket Akun AWS yang membuat Outposts memilikinya dan merupakan satu-satunya yang dapat menetapkan titik akses ke sana.

Bagian berikut menjelaskan cara membuat dan mengelola titik akses untuk bucket S3 di Outposts.

Topik

Membuat titik akses S3 di Outposts

Untuk mengakses bucket Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasi titik akses.

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan

S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

Contoh berikut menunjukkan cara membuat S3 pada jalur akses Outposts dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

> **Note**
>
> Bucket Akun AWS yang membuat Outposts memilikinya dan merupakan satu-satunya yang dapat menetapkan titik akses ke sana.

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda buatkan titik akses Outpostsnya.
4. Pilih tab titik akses Outposts.
5. Di bagian Titik akses Outposts, pilih Buat titik akses Outposts.
6. Dalam pengaturan titik akses Outposts, masukkan nama untuk titik akses, kemudian pilih cloud privat virtual (VPC) untuk titik akses.
7. Apabila Anda ingin menambahkan kebijakan untuk titik akses Anda, masukkan di bagian kebijakan titik akses Outposts.

   Untuk informasi selengkapnya, lihat Mengatur IAM dengan S3 di Outposts.

Menggunakan AWS CLI

Example

AWS CLI Contoh berikut membuat titik akses untuk bucket Outposts. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control create-access-point --account-id 123456789012
 --name example-outposts-access-point --bucket "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket" --vpc-configuration VpcId=example-vpc-12345
```

Menggunakan AWS SDK for Java

Example

Untuk contoh cara membuat titik akses untuk bucket S3 Outposts dengan AWS SDK for Java CreateOutpostsAccessPoint, lihat AWS .java di SDK for Java 2.x Code Examples.

Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts

Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Setiap kali Anda membuat titik akses untuk bucket, S3 di Outposts secara otomatis menghasilkan alias titik akses. Anda dapat menggunakan alias titik akses ini alih-alih ARN titik akses untuk operasi bidang data apa pun. Misalnya, Anda dapat menggunakan alias titik akses untuk melakukan operasi tingkat objek seperti PUT, GET, LIST, dan lainnya. Untuk mengetahui daftar operasi ini, lihat Operasi API Amazon S3 untuk mengelola objek.

Contoh berikut menunjukkan ARN dan alias titik akses untuk titik akses bernama *my-access-point*.

- ARN titik akses–`arn:aws:s3-outposts:`*region*`:`*123456789012*`:outpost/`*op-01ac5d28a6a232904*`/accesspoint/`*my-access-point*
- Alias titik akses–*my-access-po*-*o01ac5d28a6a232904e8xz5w8ijx1qzlbp3i3kuse10*`--op-s3`

Untuk informasi selengkapnya ARNs, lihat Amazon Resource Names (ARNs) di Referensi Umum AWS.

Untuk informasi selengkapnya tentang alias titik akses, lihat topik berikut.

Topik

- Alias titik akses
- Menggunakan alias titik akses dalam operasi objek S3 di Outposts
- Batasan

Alias titik akses

Alias titik akses dibuat dalam namespace yang sama dengan bucket S3 di Outposts. Saat Anda membuat titik akses, S3 di Outposts secara otomatis menghasilkan alias titik akses yang tidak dapat diubah. Alias titik akses memenuhi semua persyaratan nama bucket S3 di Outposts yang valid dan terdiri atas bagian berikut:

access point name prefix`-`*metadata*`--op-s3`

> **Note**
>
> Sufiks `--op-s3` dipesan untuk alias titik akses, kami menyarankan agar Anda tidak menggunakannya untuk nama bucket atau titik akses. Untuk informasi selengkapnya tentang aturan penamaan bucket S3 di Outposts, lihat Bekerja dengan S3 di bucket Outposts.

Menemukan alias titik akses

Contoh berikut menunjukkan cara menemukan alias titik akses menggunakan konsol Amazon S3 dan AWS CLI.

Example : Temukan dan salin alias titik akses di konsol Amazon S3

Setelah Anda membuat titik akses di konsol, Anda dapat memperoleh alias titik akses dari kolom alias Titik Akses di daftar Titik Akses.

Untuk menyalin alias titik akses

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Untuk menyalin alias titik akses, lakukan salah satu hal berikut:
   - Dalam daftar Titik Akses, pilih tombol opsi di sebelah nama titik akses, lalu pilih Salin alias Titik Akses.
   - Pilih nama titik akses. Kemudian, dalam ikhtisar titik akses Outposts, salin alias titik akses.

Example : Buat titik akses dengan menggunakan AWS CLI dan temukan alias titik akses dalam respons

AWS CLI Contoh berikut untuk `create-access-point` perintah menciptakan titik akses dan mengembalikan alias titik akses yang dihasilkan secara otomatis. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control create-access-point --bucket example-outposts-bucket --name example-
outposts-access-point --account-id 123456789012

{
    "AccessPointArn":
    "arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/
accesspoint/example-outposts-access-point",
    "Alias": "example-outp-o01ac5d28a6a232904e8xz5w8ijx1qzlbp3i3kuse10--op-s3"
}
```

Example : Dapatkan alias access point dengan menggunakan AWS CLI

AWS CLI Contoh berikut untuk `get-access-point` perintah mengembalikan informasi tentang titik akses yang ditentukan. Informasi ini termasuk alias titik akses. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control get-access-point --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --name example-outposts-access-point --account-id 123456789012

{
    "Name": "example-outposts-access-point",
    "Bucket": "example-outposts-bucket",
    "NetworkOrigin": "Vpc",
    "VpcConfiguration": {
        "VpcId": "vpc-01234567890abcdef"
    },
    "PublicAccessBlockConfiguration": {
        "BlockPublicAcls": true,
        "IgnorePublicAcls": true,
        "BlockPublicPolicy": true,
        "RestrictPublicBuckets": true
    },
    "CreationDate": "2022-09-18T17:49:15.584000+00:00",
    "Alias": "example-outp-o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3"
```

```
}
```

Example : Daftar titik akses untuk menemukan alias titik akses dengan menggunakan AWS CLI

AWS CLI Contoh berikut untuk `list-access-points` perintah mencantumkan informasi tentang titik akses yang ditentukan. Informasi ini termasuk alias titik akses. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control list-access-points --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket

{
    "AccessPointList": [
        {
            "Name": "example-outposts-access-point",
            "NetworkOrigin": "Vpc",
            "VpcConfiguration": {
                "VpcId": "vpc-01234567890abcdef"
            },
            "Bucket": "example-outposts-bucket",
            "AccessPointArn": "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point",
            "Alias": "example-outp-o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3"
        }
    ]
}
```

Menggunakan alias titik akses dalam operasi objek S3 di Outposts

Saat mengadopsi titik akses, Anda dapat menggunakan alias titik akses tanpa memerlukan perubahan kode yang ekstensif.

AWS CLI Contoh ini menunjukkan `get-object` operasi untuk bucket S3 pada Outposts. Contoh ini menggunakan alias titik akses sebagai nilai untuk `--bucket` bukan ARN titik akses penuh.

```
aws s3api get-object --bucket my-access-po-
o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3 --key  testkey  sample-object.rtf

{
```

```
    "AcceptRanges": "bytes",
    "LastModified": "2020-01-08T22:16:28+00:00",
    "ContentLength": 910,
    "ETag": "\"00751974dc146b76404bb7290f8f51bb\"",
    "VersionId": "null",
    "ContentType": "text/rtf",
    "Metadata": {}
}
```

Batasan

- Alias tidak dapat dikonfigurasi oleh pelanggan.
- Alias tidak dapat dihapus atau dimodifikasi atau dinonaktifkan pada titik akses.
- Anda tidak dapat menggunakan alias titik akses untuk operasi bidang kontrol S3 di Outposts. Untuk daftar oeprasi bidang kontrol S3 di Outposts, lihat Operasi API Amazon S3 Control untuk mengelola bucket.
- Alias tidak dapat digunakan dalam kebijakan AWS Identity and Access Management (IAM).

Melihat informasi tentang konfigurasi titik akses

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

Topik berikut menunjukkan cara mengembalikan informasi konfigurasi untuk jalur akses S3 di Outposts dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Pilih titik akses Outposts yang ingin Anda lihat detail konfigurasinya.
4. Dalam ikhtisar titik akses Outposts, tinjau detail konfigurasi titik akses.

Menggunakan AWS CLI

AWS CLI Contoh berikut mendapatkan titik akses untuk bucket Outposts. Ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control get-access-point --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mendapatkan titik akses untuk bucket Outposts.

```
import com.amazonaws.services.s3control.model.*;

public void getAccessPoint(String accessPointArn) {

    GetAccessPointRequest reqGetAP = new GetAccessPointRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn);

    GetAccessPointResult respGetAP = s3ControlClient.getAccessPoint(reqGetAP);
    System.out.printf("GetAccessPoint Response: %s%n", respGetAP.toString());

}
```

Melihat daftar titik akses Amazon S3 di Outposts Anda

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

Topik berikut menunjukkan cara mengembalikan daftar S3 Anda di jalur akses Outposts dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/

2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Dalam titik akses Outposts, tinjau daftar titik akses S3 di Outposts Anda.

Menggunakan AWS CLI

AWS CLI Contoh berikut mencantumkan titik akses untuk bucket Outposts. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control list-access-points --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mencantumkan titik akses untuk bucket Outposts.

```
import com.amazonaws.services.s3control.model.*;

public void listAccessPoints(String bucketArn) {

    ListAccessPointsRequest reqListAPs = new ListAccessPointsRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    ListAccessPointsResult respListAPs = s3ControlClient.listAccessPoints(reqListAPs);
    System.out.printf("ListAccessPoints Response: %s%n", respListAPs.toString());

}
```

Menghapus titik akses

Titik akses menyederhanakan pengelolaan akses data dalam skala besar untuk set data bersama di Amazon S3. Titik akses diberi nama titik akhir jaringan yang melekat ke bucket yang dapat Anda gunakan untuk melakukan operasi objek Amazon S3, seperti `GetObject` dan `PutObject`. Dengan S3 di Outposts, Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Titik akses hanya mendukung virtual-host-style pengalamatan.

Contoh berikut menunjukkan cara menghapus titik akses dengan menggunakan AWS Management Console dan AWS Command Line Interface (AWS CLI).

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Di bagian titik akses Outposts, pilih titik akses Outposts yang ingin Anda hapus.
4. Pilih Hapus.
5. Konfirmasi penghapusan.

Menggunakan AWS CLI

AWS CLI Contoh berikut menghapus titik akses Outposts. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control delete-access-point --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Menambahkan atau mengedit kebijakan titik akses

Titik akses memiliki izin dan kendali jaringan yang berbeda yang diterapkan Amazon S3 di Outposts untuk setiap permintaan yang dibuat melalui titik akses tersebut. Setiap titik akses memberlakukan kebijakan titik akses khusus yang bekerja bersama kebijakan bucket yang melekat pada bucket dasar. Untuk informasi selengkapnya, lihat Titik Akses.

Topik berikut menunjukkan cara menambahkan atau mengedit kebijakan jalur akses untuk jalur akses S3 di Outposts Anda dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda edit kebijakan titik aksesnya.
4. Pilih tab titik akses Outposts.
5. Di bagian titik akses Outposts, pilih titik akses yang kebijakannya ingin Anda edit, lalu pilih Edit kebijakan.

6. Tambahkan atau edit kebijakan di bagian kebijakan titik akses Outposts. Untuk informasi selengkapnya, lihat Mengatur IAM dengan S3 di Outposts.

Menggunakan AWS CLI

AWS CLI Contoh berikut menempatkan kebijakan pada titik akses Outposts.

1. Simpan kebijakan titik akses berikut ke file JSON. Dalam contoh ini, file dinamai `appolicy1.json`. Ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
{
   "Version":"2012-10-17",
   "Id":"exampleAccessPointPolicy",
   "Statement":[
      {
         "Sid":"st1",
         "Effect":"Allow",
         "Principal":{
            "AWS":"123456789012"
         },
         "Action":"s3-outposts:*",
         "Resource":"arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point
      }
   ]
}
```

2. Kirim file JSON sebagai bagian dari perintah CLI `put-access-point-policy`. Ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control put-access-point-policy --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --policy file://appolicy1.json
```

Menggunakan AWS SDK for Java

Contoh SDK untuk Java berikut menempatkan kebijakan pada titik akses Outposts.

```
import com.amazonaws.services.s3control.model.*;
```

```
public void putAccessPointPolicy(String accessPointArn) {

    String policy = "{\"Version\":\"2012-10-17\",\"Id\":\"testAccessPointPolicy\",
\"Statement\":[{\"Sid\":\"st1\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"" +
 AccountId + "\"},\"Action\":\"s3-outposts:*\",\"Resource\":\"" + accessPointArn +
 "\"}]}";

    PutAccessPointPolicyRequest reqPutAccessPointPolicy = new
 PutAccessPointPolicyRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn)
            .withPolicy(policy);

    PutAccessPointPolicyResult respPutAccessPointPolicy =
 s3ControlClient.putAccessPointPolicy(reqPutAccessPointPolicy);
    System.out.printf("PutAccessPointPolicy Response: %s%n",
 respPutAccessPointPolicy.toString());
    printWriter.printf("PutAccessPointPolicy Response: %s%n",
 respPutAccessPointPolicy.toString());

}
```

Melihat kebijakan titik akses untuk titik akses S3 di Outposts

Titik akses memiliki izin dan kendali jaringan yang berbeda yang diterapkan Amazon S3 di Outposts untuk setiap permintaan yang dibuat melalui titik akses tersebut. Setiap titik akses memberlakukan kebijakan titik akses khusus yang bekerja bersama kebijakan bucket yang melekat pada bucket dasar. Untuk informasi selengkapnya, lihat Titik Akses.

Untuk informasi selengkapnya tentang bekerja dengan titik akses dalam S3 di Outposts, lihat Bekerja dengan S3 di bucket Outposts.

Topik berikut menunjukkan cara melihat kebijakan jalur akses S3 di Outposts dengan menggunakan AWS Management Console AWS Command Line Interface ,AWS CLI(), dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Pilih titik akses Outposts yang ingin Anda lihat kebijakannya.
4. Pada tab Izin, tinjau kebijakan titik akses S3 di Outposts.

5. Untuk mengedit kebijakan titik akses, lihat Menambahkan atau mengedit kebijakan titik akses.

Menggunakan AWS CLI

AWS CLI Contoh berikut mendapatkan kebijakan untuk jalur akses Outposts. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control get-access-point-policy --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mendapatkan kebijakan untuk titik akses Outposts.

```
import com.amazonaws.services.s3control.model.*;

public void getAccessPointPolicy(String accessPointArn) {

    GetAccessPointPolicyRequest reqGetAccessPointPolicy = new
 GetAccessPointPolicyRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn);

    GetAccessPointPolicyResult respGetAccessPointPolicy =
 s3ControlClient.getAccessPointPolicy(reqGetAccessPointPolicy);
    System.out.printf("GetAccessPointPolicy Response: %s%n",
 respGetAccessPointPolicy.toString());
    printWriter.printf("GetAccessPointPolicy Response: %s%n",
 respGetAccessPointPolicy.toString());

}
```

Bekerja dengan titik akhir Amazon S3 di Outposts

Untuk merutekan permintaan ke titik akses Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Untuk membuat titik akhir, Anda akan memerlukan koneksi aktif dengan tautan layanan ke wilayah asal Outposts Anda. Setiap Cloud Privat Virtual (VPC) pada Outposts Anda dapat memiliki satu titik akhir terkait. Untuk informasi selengkapnya

tentang kuota titik akhir, lihat Persyaratan jaringan S3 di Outposts. Anda harus membuat titik akhir agar dapat mengakses bucket Outposts Anda dan melakukan operasi objek. Untuk informasi selengkapnya, lihat Titik akhir.

Setelah Anda membuat titik akhir, Anda dapat menggunakan bidang 'Status', untuk memahami status titik akhir. Apabila Outposts Anda offline, mereka akan mengembalikan hasil CREATE_FAILED. Anda dapat memeriksa koneksi tautan layanan, menghapus titik akhir, dan mencoba lagi operasi pembuatan setelah koneksi Anda dilanjutkan. Untuk daftar kode kesalahan tambahan, lihat di bawah. Untuk informasi selengkapnya, lihat Titik akhir.

API	Status	Kode Kesalahan Alasan Gagal	Pesan-Alasan Gagal
`CreateEndpoint`	Create_Failed	OutpostNotReachable	Titik akhir tidak dapat dibuat karena koneksi tautan layanan ke Wilayah asal Outposts Anda gagal. Periksa koneksi Anda, hapus titik akhir, dan coba lagi.
`CreateEndpoint`	Create_Failed	InternalError	Titik akhir tidak dapat dibuat karena Kesalahan Internal. Hapus titik akhir dan buat lagi.
`DeleteEndpoint`	Delete_Failed	OutpostNotReachable	Titik akhir tidak dapat dihapus karena koneksi tautan layanan ke Wilayah asal Outposts Anda gagal. Periksa koneksi Anda dan coba lagi.
`DeleteEndpoint`	Delete_Failed	InternalError	Titik akhir tidak dapat dihapus karena Kesalahan Internal. Coba lagi.

Untuk informasi selengkapnya tentang bekerja dengan bucket pada S3 di Outposts, lihat Bekerja dengan S3 di bucket Outposts.

Bagian berikut menjelaskan cara membuat dan mengelola titik akhir untuk S3 di Outposts.

Topik

- Membuat titik akhir di Outpost
- Melihat daftar titik akhir Amazon S3 di Outposts Anda

- Menghapus titik akhir Amazon S3 di Outposts

Membuat titik akhir di Outpost

Untuk merutekan permintaan ke titik akses Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Untuk membuat titik akhir, Anda akan memerlukan koneksi aktif dengan tautan layanan ke wilayah asal Outposts Anda. Setiap Cloud Privat Virtual (VPC) pada Outposts Anda dapat memiliki satu titik akhir terkait. Untuk informasi selengkapnya tentang kuota titik akhir, lihat Persyaratan jaringan S3 di Outposts. Anda harus membuat titik akhir agar dapat mengakses bucket Outposts Anda dan melakukan operasi objek. Untuk informasi selengkapnya, lihat Titik akhir.

Izin

Untuk informasi selengkapnya tentang izin yang diperlukan untuk membuat titik akhir, lihat Izin untuk titik akhir S3 di Outposts.

Saat Anda membuat titik akhir, S3 di Outposts juga membuat peran yang terkait dengan layanan dalam Akun AWS Anda. Untuk informasi selengkapnya, lihat Menggunakan Peran Terkait Layanan untuk Amazon S3 di Outposts.

Contoh berikut menunjukkan cara membuat S3 pada titik akhir Outposts dengan menggunakan AWS Command Line Interface ,AWS CLI(), AWS Management Console dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Pilih tab titik akhir Outposts.
4. Pilih Buat titik akhir Outposts.
5. Dalam Outpost, pilih Outpost tempat titik akhir ini akan dibuat.
6. Dalam VPC, pilih VPC yang belum memiliki titik akhir dan yang juga sesuai dengan aturan untuk titik akhir Outposts.

   Virtual Private Cloud (VPC) memungkinkan Anda meluncurkan AWS sumber daya ke jaringan virtual yang Anda tentukan. Jaringan virtual ini sangat mirip dengan jaringan tradisional yang

akan Anda operasikan di pusat data Anda sendiri, dengan manfaat dari penggunaan infrastruktur AWS yang dapat diskalakan.

Jika Anda tidak memiliki VPC, pilih Buat VPC. Untuk informasi selengkapnya, lihat Membuat titik akses yang dibatasi untuk virtual private cloud (VPC) di Panduan Pengguna Amazon S3.

7. Pilih Buat titik akhir Outposts.

Menggunakan AWS CLI

Example

AWS CLI Contoh berikut membuat endpoint untuk Outpost dengan menggunakan jenis akses sumber daya VPC. VPC berasal dari subnet. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1
```

AWS CLI Contoh berikut membuat endpoint untuk Outpost dengan menggunakan jenis akses pool alamat IP (CoIP pool) milik pelanggan. Untuk menjalankan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1 --access-type CustomerOwnedIp --
customer-owned-ipv4-pool ipv4pool-coip-12345678901234567
```

Menggunakan AWS SDK for Java

Example

Untuk contoh cara membuat endpoint untuk Outpost S3 dengan SDK for Java, CreateOutpostsEndPointlihat AWS .java di SDK for Java 2.x Code Examples.AWS

Melihat daftar titik akhir Amazon S3 di Outposts Anda

Untuk merutekan permintaan ke titik akses Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Untuk membuat titik akhir, Anda akan memerlukan koneksi aktif dengan tautan layanan ke wilayah asal Outposts Anda. Setiap Cloud Privat Virtual (VPC) pada Outposts Anda dapat memiliki satu titik akhir terkait. Untuk informasi selengkapnya

tentang kuota titik akhir, lihat Persyaratan jaringan S3 di Outposts. Anda harus membuat titik akhir agar dapat mengakses bucket Outposts Anda dan melakukan operasi objek. Untuk informasi selengkapnya, lihat Titik akhir.

Contoh berikut menunjukkan cara mengembalikan daftar S3 Anda di titik akhir Outposts dengan menggunakan AWS Command Line Interface ,AWS CLI(), AWS Management Console dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Pada halaman titik akses Outposts, pilih tab titik akhir Outposts.
4. Dalam titik akhir Outposts, Anda dapat melihat daftar titik akhir S3 di Outposts.

Menggunakan AWS CLI

AWS CLI Contoh berikut mencantumkan titik akhir untuk AWS Outposts sumber daya yang terkait dengan akun Anda. Untuk informasi selengkapnya tentang perintah ini, lihat daftar-titik akhir dalam Referensi AWS CLI .

```
aws s3outposts list-endpoints
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut mencantumkan titik akhir untuk Outposts. Untuk informasi selengkapnya, lihat ListEndpoints di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.services.s3outposts.AmazonS3Outposts;
import com.amazonaws.services.s3outposts.AmazonS3OutpostsClientBuilder;
import com.amazonaws.services.s3outposts.model.ListEndpointsRequest;
import com.amazonaws.services.s3outposts.model.ListEndpointsResult;

public void listEndpoints() {
    AmazonS3Outposts s3OutpostsClient = AmazonS3OutpostsClientBuilder
                .standard().build();

    ListEndpointsRequest listEndpointsRequest = new ListEndpointsRequest();
    ListEndpointsResult listEndpointsResult =
 s3OutpostsClient.listEndpoints(listEndpointsRequest);
```

```
    System.out.println("List endpoints result is " + listEndpointsResult);
}
```

Menghapus titik akhir Amazon S3 di Outposts

Untuk merutekan permintaan ke titik akses Amazon S3 di Outposts, Anda harus membuat dan mengonfigurasikan titik akhir S3 di Outposts. Untuk membuat titik akhir, Anda akan memerlukan koneksi aktif dengan tautan layanan ke wilayah asal Outposts Anda. Setiap Cloud Privat Virtual (VPC) pada Outposts Anda dapat memiliki satu titik akhir terkait. Untuk informasi selengkapnya tentang kuota titik akhir, lihat Persyaratan jaringan S3 di Outposts. Anda harus membuat titik akhir agar dapat mengakses bucket Outposts Anda dan melakukan operasi objek. Untuk informasi selengkapnya, lihat Titik akhir.

Contoh berikut menunjukkan cara menghapus S3 Anda pada titik akhir Outposts dengan menggunakan AWS Command Line Interface ,AWS CLI(), AWS Management Console dan. AWS SDK untuk Java

Menggunakan konsol S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih titik akses Outposts.
3. Pada halaman titik akses Outposts, pilih tab titik akhir Outposts.
4. Dalam titik akhir Outposts, pilih titik akhir yang ingin Anda hapus, lalu pilih Hapus.

Menggunakan AWS CLI

AWS CLI Contoh berikut menghapus endpoint untuk Outpost. Untuk menjalankan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3outposts delete-endpoint --endpoint-id example-endpoint-id --outpost-
id op-01ac5d28a6a232904
```

Menggunakan AWS SDK for Java

Contoh SDK for Java berikut menghapus titik akhir untuk Outpost. Untuk menggunakan contoh ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
import com.amazonaws.arn.Arn;
```

```
import com.amazonaws.services.s3outposts.AmazonS3Outposts;
import com.amazonaws.services.s3outposts.AmazonS3OutpostsClientBuilder;
import com.amazonaws.services.s3outposts.model.DeleteEndpointRequest;

public void deleteEndpoint(String endpointArnInput) {
    String outpostId = "op-01ac5d28a6a232904";
    AmazonS3Outposts s3OutpostsClient = AmazonS3OutpostsClientBuilder
                .standard().build();

    Arn endpointArn = Arn.fromString(endpointArnInput);
    String[] resourceParts = endpointArn.getResource().getResource().split("/");
    String endpointId = resourceParts[resourceParts.length - 1];
    DeleteEndpointRequest deleteEndpointRequest = new DeleteEndpointRequest()
                .withEndpointId(endpointId)
                .withOutpostId(outpostId);
    s3OutpostsClient.deleteEndpoint(deleteEndpointRequest);
    System.out.println("Endpoint with id " + endpointId + " is deleted.");
}
```

Bekerja dengan S3 di objek Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API.

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Object ARNs menggunakan format berikut, yang mencakup Outpost homed ke, Akun AWS ID, Outpost ID, nama bucket, dan kunci objek: Wilayah AWS

```
arn:aws:s3-outposts:us-west-2:123456789012:outpost/ op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket1/object/myobject
```

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region,

Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Topik

Unggah objek ke ember S3 di Outposts

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Berikut AWS CLI dan AWS SDK untuk Java contoh menunjukkan cara mengunggah objek ke bucket S3 di Outposts dengan menggunakan titik akses.

AWS CLI

Example

Contoh berikut menempatkan objek bernama `sample-object.xml` ke bucket S3 di Outposts (`s3-outposts:PutObject`) menggunakan AWS CLI. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat put-object dalam Referensi AWS CLI .

```
aws s3api put-object --bucket arn:aws:s3-
outposts:Region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --key sample-object.xml --body sample-object.xml
```

SDK for Java

Example

Untuk contoh cara mengunggah objek ke bucket Outposts S3 dengan AWS SDK for Java PutObjectOnOutpost, lihat .java di Contoh Kode SDK AWS for Java 2.x.

Menyalin objek di bucket Amazon S3 di Outposts menggunakan AWS SDK untuk Java

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Contoh berikut menunjukkan kepada Anda cara menyalin objek dalam bucket S3 di Outposts menggunakan AWS SDK untuk Java.

Menggunakan AWS SDK for Java

Contoh S3 di Outposts berikut menyalin objek ke objek baru di bucket yang sama menggunakan SDK for Java. Untuk menggunakan contoh ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.CopyObjectRequest;

public class CopyObject {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
        String sourceKey = "*** Source object key ***";
        String destinationKey = "*** Destination object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
```

```
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Copy the object into a new object in the same bucket.
            CopyObjectRequest copyObjectRequest = new CopyObjectRequest(accessPointArn,
 sourceKey, accessPointArn, destinationKey);
            s3Client.copyObject(copyObjectRequest);
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Mendapatkan objek dari bucket Amazon S3 di Outposts

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek

Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Contoh berikut menunjukkan kepada Anda cara mengunduh (mendapatkan) objek menggunakan AWS Command Line Interface (AWS CLI) dan AWS SDK untuk Java.

Menggunakan AWS CLI

Contoh berikut mendapatkan objek bernama `sample-object.xml` dari bucket S3 di Outposts (`s3-outposts:GetObject`) menggunakan AWS CLI. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat get-object dalam Referensi AWS CLI .

```
aws s3api get-object --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point --key testkey sample-object.xml
```

Menggunakan AWS SDK for Java

Contoh S3 pada Outposts berikut mendapatkan objek menggunakan SDK for Java. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya, lihat GetObject di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.GetObjectRequest;
import com.amazonaws.services.s3.model.ResponseHeaderOverrides;
import com.amazonaws.services.s3.model.S3Object;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;

public class GetObject {
```

```
    public static void main(String[] args) throws IOException {
        String accessPointArn = "*** access point ARN ***";
        String key = "*** Object key ***";

        S3Object fullObject = null, objectPortion = null, headerOverrideObject = null;
        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Get an object and print its contents.
            System.out.println("Downloading an object");
            fullObject = s3Client.getObject(new GetObjectRequest(accessPointArn, key));
            System.out.println("Content-Type: " +
 fullObject.getObjectMetadata().getContentType());
            System.out.println("Content: ");
            displayTextInputStream(fullObject.getObjectContent());

            // Get a range of bytes from an object and print the bytes.
            GetObjectRequest rangeObjectRequest = new GetObjectRequest(accessPointArn,
 key)
                    .withRange(0, 9);
            objectPortion = s3Client.getObject(rangeObjectRequest);
            System.out.println("Printing bytes retrieved.");
            displayTextInputStream(objectPortion.getObjectContent());

            // Get an entire object, overriding the specified response headers, and
 print the object's content.
            ResponseHeaderOverrides headerOverrides = new ResponseHeaderOverrides()
                    .withCacheControl("No-cache")
                    .withContentDisposition("attachment; filename=example.txt");
            GetObjectRequest getObjectRequestHeaderOverride = new
 GetObjectRequest(accessPointArn, key)
                    .withResponseHeaders(headerOverrides);
            headerOverrideObject = s3Client.getObject(getObjectRequestHeaderOverride);
            displayTextInputStream(headerOverrideObject.getObjectContent());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
```

```
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        } finally {
            // To ensure that the network connection doesn't remain open, close any
 open input streams.
            if (fullObject != null) {
                fullObject.close();
            }
            if (objectPortion != null) {
                objectPortion.close();
            }
            if (headerOverrideObject != null) {
                headerOverrideObject.close();
            }
        }
    }

    private static void displayTextInputStream(InputStream input) throws IOException {
        // Read the text input stream one line at a time and display each line.
        BufferedReader reader = new BufferedReader(new InputStreamReader(input));
        String line = null;
        while ((line = reader.readLine()) != null) {
            System.out.println(line);
        }
        System.out.println();
    }
}
```

Membuat daftar objek di Amazon S3 pada bucket Outposts

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

> **Note**
>
> Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Contoh berikut menunjukkan cara membuat daftar objek dalam bucket S3 di Outposts menggunakan AWS CLI dan. AWS SDK untuk Java

Menggunakan AWS CLI

Contoh berikut mencantumkan objek dalam bucket S3 pada Outposts `s3-outposts:ListObjectsV2` () dengan menggunakan. AWS CLI Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat list-objects-v2 di AWS CLI Referensi.

```
aws s3api list-objects-v2 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-access-point
```

> **Note**
>
> Saat menggunakan tindakan ini dengan Amazon S3 di Outposts melalui AWS SDKs, Anda memberikan titik akses Outposts ARN sebagai pengganti nama bucket, dalam bentuk berikut:. `arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/`

`accesspoint/example-Outposts-Access-Point` Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Menggunakan AWS SDK for Java

Contoh S3 di Outposts berikut membuat daftar objek di bucket dengan menggunakan SDK untuk Java. Untuk menggunakan contoh ini, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

> **Important**
>
> Contoh ini menggunakan ListObjectsV2, yang merupakan revisi terbaru dari operasi `ListObjects` API. Kami menyarankan agar Anda menggunakan operasi API yang direvisi ini untuk pengembangan aplikasi. Untuk kompatibilitas mundur, Amazon S3 terus mendukung versi sebelumnya dari operasi API ini.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.ListObjectsV2Request;
import com.amazonaws.services.s3.model.ListObjectsV2Result;
import com.amazonaws.services.s3.model.S3ObjectSummary;

public class ListObjectsV2 {

    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            System.out.println("Listing objects");
```

```
            // maxKeys is set to 2 to demonstrate the use of
            // ListObjectsV2Result.getNextContinuationToken()
            ListObjectsV2Request req = new
 ListObjectsV2Request().withBucketName(accessPointArn).withMaxKeys(2);
            ListObjectsV2Result result;

            do {
                result = s3Client.listObjectsV2(req);

                for (S3ObjectSummary objectSummary : result.getObjectSummaries()) {
                    System.out.printf(" - %s (size: %d)\n", objectSummary.getKey(),
 objectSummary.getSize());
                }
                // If there are more than maxKeys keys in the bucket, get a
 continuation token
                // and list the next objects.
                String token = result.getNextContinuationToken();
                System.out.println("Next Continuation Token: " + token);
                req.setContinuationToken(token);
            } while (result.isTruncated());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Menghapus objek di bucket Amazon S3 di Outposts

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Contoh berikut menunjukkan cara menghapus satu objek atau beberapa objek di bucket S3 di Outposts dengan menggunakan AWS CLI() AWS Command Line Interface dan. AWS SDK untuk Java

Menggunakan AWS CLI

Contoh berikut menunjukkan kepada Anda cara menghapus satu atau beberapa objek dalam bucket S3 di Outposts.

delete-object

Contoh berikut menghapus objek bernama `sample-object.xml` dari bucket S3 di Outposts (`s3-outposts:DeleteObject`) menggunakan AWS CLI. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat delete-object dalam Referensi AWS CLI .

```
aws s3api delete-object --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --key sample-object.xml
```

delete-objects

Contoh berikut menghapus dua objek bernama `sample-object.xml` dan `test1.text` dari bucket S3 di Outposts (`s3-outposts:DeleteObject`) menggunakan AWS CLI. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat delete-objects dalam Referensi AWS CLI .

```
aws s3api delete-objects --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --delete file://delete.json

delete.json
{
  "Objects": [
    {
      "Key": "test1.txt"
    },
    {
      "Key": "sample-object.xml"
    }
  ],
  "Quiet": false
}
```

Menggunakan AWS SDK for Java

Contoh berikut menunjukkan kepada Anda cara menghapus satu atau beberapa objek dalam bucket S3 di Outposts.

DeleteObject

Contoh S3 di Outposts berikut menghapus objek dalam bucket menggunakan SDK for Java. Untuk menggunakan contoh ini, tentukan ARN titik akses untuk Outpost dan nama kunci untuk objek yang ingin Anda hapus. Untuk informasi selengkapnya, lihat DeleteObject dalam Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
```

```
import com.amazonaws.services.s3.model.DeleteObjectRequest;

public class DeleteObject {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
        String keyName = "*** key name ****";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            s3Client.deleteObject(new DeleteObjectRequest(accessPointArn, keyName));
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

DeleteObjects

Contoh S3 di Outposts berikut mengunggah lalu menghapus objek dalam bucket menggunakan SDK for Java. Untuk menggunakan contoh ini, tentukan ARN titik akses untuk Outpost. Untuk informasi selengkapnya, lihat DeleteObjects di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.DeleteObjectsRequest;
import com.amazonaws.services.s3.model.DeleteObjectsRequest.KeyVersion;
import com.amazonaws.services.s3.model.DeleteObjectsResult;

import java.util.ArrayList;
```

```
public class DeleteObjects {

    public static void main(String[] args) {
       String accessPointArn = "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point";


        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Upload three sample objects.
            ArrayList<KeyVersion> keys = new ArrayList<KeyVersion>();
            for (int i = 0; i < 3; i++) {
                String keyName = "delete object example " + i;
                s3Client.putObject(accessPointArn, keyName, "Object number " + i + "
 to be deleted.");
                keys.add(new KeyVersion(keyName));
            }
            System.out.println(keys.size() + " objects successfully created.");

            // Delete the sample objects.
            DeleteObjectsRequest multiObjectDeleteRequest = new
 DeleteObjectsRequest(accessPointArn)
                    .withKeys(keys)
                    .withQuiet(false);

            // Verify that the objects were deleted successfully.
            DeleteObjectsResult delObjRes =
 s3Client.deleteObjects(multiObjectDeleteRequest);
            int successfulDeletes = delObjRes.getDeletedObjects().size();
            System.out.println(successfulDeletes + " objects successfully
 deleted.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
```

```
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Menggunakan HeadBucket untuk menentukan apakah bucket S3 di Outposts ada dan Anda memiliki izin akses

Objek adalah entitas dasar yang disimpan di Amazon S3 di Outposts. Setiap objek dimuat dalam bucket. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat menentukan bucket untuk operasi objek, Anda menggunakan Amazon Resource Name (ARN) titik akses atau alias titik akses. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Contoh berikut menunjukkan format ARN untuk S3 pada jalur akses Outposts, yang mencakup Wilayah AWS kode untuk Wilayah tempat Pos Luar berada, ID, Akun AWS ID Pos Luar, dan nama titik akses:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Untuk informasi lebih lanjut tentang S3 di ARNs Outposts, lihat. Sumber daya ARNs untuk S3 di Outposts

> **Note**
>
> Dengan Amazon S3 di Outposts, data objek selalu disimpan di Outpost. Saat AWS memasang rak Outpost, data Anda tetap lokal di Outpost Anda untuk memenuhi persyaratan residensi data. Objek Anda tidak akan meninggalkan Outpost dan tidak berada di Wilayah AWS. Karena di-host In-region, Anda tidak dapat menggunakan konsol untuk mengunggah atau mengelola objek di Outpost Anda. AWS Management Console Namun, Anda dapat menggunakan REST API, AWS Command Line Interface (AWS CLI), dan AWS SDKs untuk mengunggah dan mengelola objek Anda melalui titik akses Anda.

Berikut AWS Command Line Interface (AWS CLI) dan AWS SDK untuk Java contoh menunjukkan cara menggunakan operasi HeadBucket API untuk menentukan apakah bucket Amazon S3 di Outposts ada dan apakah Anda memiliki izin untuk mengaksesnya. Untuk informasi selengkapnya, lihat HeadBucket dalam Referensi API Amazon Simple Storage Service.

Menggunakan AWS CLI

Contoh S3 di AWS CLI Outposts berikut menggunakan `head-bucket` perintah untuk menentukan apakah ada bucket dan Anda memiliki izin untuk mengaksesnya. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat head-bucket dalam Referensi AWS CLI .

```
aws s3api head-bucket --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Menggunakan AWS SDK for Java

Contoh S3 di Outposts berikut menujukkan cara menentukan jika terdapat bucket dan jika Anda memiliki izin untuk mengaksesnya. Untuk menggunakan contoh ini, tentukan ARN titik akses untuk Outpost. Untuk informasi selengkapnya, lihat HeadBucket di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.HeadBucketRequest;

public class HeadBucket {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();
```

```
            s3Client.headBucket(new HeadBucketRequest(accessPointArn));
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Melakukan dan mengelola unggahan multipart dengan SDK for Java

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts sumber daya serta menyimpan serta mengambil objek lokal untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan S3 di Outposts dengan AWS SDK untuk Java untuk melakukan dan mengelola unggahan multipart.

Topik

- Lakukan unggahan multipart di bucket S3 di Outposts
- Salin objek besar di bucket S3 di Outposts dengan menggunakan unggahan multibagian
- Buat daftar bagian objek di bucket S3 di Outposts
- Dapatkan daftar unggahan multibagian yang sedang berlangsung di bucket S3 di Outposts

Lakukan unggahan multipart di bucket S3 di Outposts

Contoh S3 di Outposts berikut memulai, mengunggah, dan menyelesaikan unggahan multibagian objek ke bucket dengan menggunakan SDK for Java. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya,

lihat Mengunggah objek menggunakan unggahan multibagian di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.ArrayList;
import java.util.List;

public class MultipartUploadCopy {
    public static void main(String[] args) {
        String accessPointArn = "*** Source access point ARN ***";
        String sourceObjectKey = "*** Source object key ***";
        String destObjectKey = "*** Target object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Initiate the multipart upload.
            InitiateMultipartUploadRequest initRequest = new
 InitiateMultipartUploadRequest(accessPointArn, destObjectKey);
            InitiateMultipartUploadResult initResult =
 s3Client.initiateMultipartUpload(initRequest);

            // Get the object size to track the end of the copy operation.
            GetObjectMetadataRequest metadataRequest = new
 GetObjectMetadataRequest(accessPointArn, sourceObjectKey);
            ObjectMetadata metadataResult =
 s3Client.getObjectMetadata(metadataRequest);
            long objectSize = metadataResult.getContentLength();

            // Copy the object using 5 MB parts.
            long partSize = 5 * 1024 * 1024;
            long bytePosition = 0;
            int partNum = 1;
```

```
            List<CopyPartResult> copyResponses = new ArrayList<CopyPartResult>();
            while (bytePosition < objectSize) {
                // The last part might be smaller than partSize, so check to make sure
                // that lastByte isn't beyond the end of the object.
                long lastByte = Math.min(bytePosition + partSize - 1, objectSize - 1);

                // Copy this part.
                CopyPartRequest copyRequest = new CopyPartRequest()
                        .withSourceBucketName(accessPointArn)
                        .withSourceKey(sourceObjectKey)
                        .withDestinationBucketName(accessPointArn)
                        .withDestinationKey(destObjectKey)
                        .withUploadId(initResult.getUploadId())
                        .withFirstByte(bytePosition)
                        .withLastByte(lastByte)
                        .withPartNumber(partNum++);
                copyResponses.add(s3Client.copyPart(copyRequest));
                bytePosition += partSize;
            }

            // Complete the upload request to concatenate all uploaded parts and make
the copied object available.
            CompleteMultipartUploadRequest completeRequest = new
CompleteMultipartUploadRequest(
                    accessPointArn,
                    destObjectKey,
                    initResult.getUploadId(),
                    getETags(copyResponses));
            s3Client.completeMultipartUpload(completeRequest);
            System.out.println("Multipart copy complete.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }

    // This is a helper function to construct a list of ETags.
    private static List<PartETag> getETags(List<CopyPartResult> responses) {
        List<PartETag> etags = new ArrayList<PartETag>();
```

```
        for (CopyPartResult response : responses) {
            etags.add(new PartETag(response.getPartNumber(), response.getETag()));
        }
        return etags;
    }
```

Salin objek besar di bucket S3 di Outposts dengan menggunakan unggahan multibagian

Contoh S3 di Outposts berikut menggunakan SDK for Java untuk menyalin objek di bucket. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.ArrayList;
import java.util.List;

public class MultipartUploadCopy {
    public static void main(String[] args) {
        String accessPointArn = "*** Source access point ARN ***";
        String sourceObjectKey = "*** Source object key ***";
        String destObjectKey = "*** Target object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Initiate the multipart upload.
            InitiateMultipartUploadRequest initRequest = new
 InitiateMultipartUploadRequest(accessPointArn, destObjectKey);
            InitiateMultipartUploadResult initResult =
 s3Client.initiateMultipartUpload(initRequest);
```

```
            // Get the object size to track the end of the copy operation.
            GetObjectMetadataRequest metadataRequest = new
GetObjectMetadataRequest(accessPointArn, sourceObjectKey);
            ObjectMetadata metadataResult =
s3Client.getObjectMetadata(metadataRequest);
            long objectSize = metadataResult.getContentLength();

            // Copy the object using 5 MB parts.
            long partSize = 5 * 1024 * 1024;
            long bytePosition = 0;
            int partNum = 1;
            List<CopyPartResult> copyResponses = new ArrayList<CopyPartResult>();
            while (bytePosition < objectSize) {
                // The last part might be smaller than partSize, so check to make sure
                // that lastByte isn't beyond the end of the object.
                long lastByte = Math.min(bytePosition + partSize - 1, objectSize - 1);

                // Copy this part.
                CopyPartRequest copyRequest = new CopyPartRequest()
                        .withSourceBucketName(accessPointArn)
                        .withSourceKey(sourceObjectKey)
                        .withDestinationBucketName(accessPointArn)
                        .withDestinationKey(destObjectKey)
                        .withUploadId(initResult.getUploadId())
                        .withFirstByte(bytePosition)
                        .withLastByte(lastByte)
                        .withPartNumber(partNum++);
                copyResponses.add(s3Client.copyPart(copyRequest));
                bytePosition += partSize;
            }

            // Complete the upload request to concatenate all uploaded parts and make
the copied object available.
            CompleteMultipartUploadRequest completeRequest = new
CompleteMultipartUploadRequest(
                    accessPointArn,
                    destObjectKey,
                    initResult.getUploadId(),
                    getETags(copyResponses));
            s3Client.completeMultipartUpload(completeRequest);
            System.out.println("Multipart copy complete.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
```

```
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }

    // This is a helper function to construct a list of ETags.
    private static List<PartETag> getETags(List<CopyPartResult> responses) {
        List<PartETag> etags = new ArrayList<PartETag>();
        for (CopyPartResult response : responses) {
            etags.add(new PartETag(response.getPartNumber(), response.getETag()));
        }
        return etags;
    }
}
```

Buat daftar bagian objek di bucket S3 di Outposts

Contoh S3 di Outposts berikut mencantumkan bagian objek di bucket dengan menggunakan SDK for Java. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.List;

public class ListParts {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
        String keyName = "*** Key name ***";
        String uploadId = "*** Upload ID ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
```

```
                    .enableUseArnRegion()
                    .build();

            ListPartsRequest listPartsRequest = new ListPartsRequest(accessPointArn,
 keyName, uploadId);
            PartListing partListing = s3Client.listParts(listPartsRequest);
            List<PartSummary> partSummaries = partListing.getParts();

            System.out.println(partSummaries.size() + " multipart upload parts");
            for (PartSummary p : partSummaries) {
                System.out.println("Upload part: Part number = \"" + p.getPartNumber()
 + "\", ETag = " + p.getETag());
            }

        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Dapatkan daftar unggahan multibagian yang sedang berlangsung di bucket S3 di Outposts

Contoh S3 di Outposts berikut ini menunjukkan cara mengambil daftar unggahan multibagian yang sedang berlangsung dari bucket Outposts dengan menggunakan SDK for Java. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.ListMultipartUploadsRequest;
import com.amazonaws.services.s3.model.MultipartUpload;
import com.amazonaws.services.s3.model.MultipartUploadListing;

import java.util.List;
```

```
public class ListMultipartUploads {
    public static void main(String[] args) {
                String accessPointArn = "*** access point ARN ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Retrieve a list of all in-progress multipart uploads.
            ListMultipartUploadsRequest allMultipartUploadsRequest = new
 ListMultipartUploadsRequest(accessPointArn);
            MultipartUploadListing multipartUploadListing =
 s3Client.listMultipartUploads(allMultipartUploadsRequest);
            List<MultipartUpload> uploads =
 multipartUploadListing.getMultipartUploads();

            // Display information about all in-progress multipart uploads.
            System.out.println(uploads.size() + " multipart upload(s) in progress.");
            for (MultipartUpload u : uploads) {
                System.out.println("Upload in progress: Key = \"" + u.getKey() + "\",
 id = " + u.getUploadId());
            }
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Menggunakan presigned URLs untuk S3 di Outposts

Untuk memberikan akses terbatas waktu ke objek yang disimpan secara lokal di Outpost tanpa memperbarui kebijakan bucket, Anda dapat menggunakan URL yang telah ditetapkan sebelumnya.

Dengan presigned URLs, Anda sebagai pemilik bucket dapat berbagi objek dengan individu di virtual private cloud (VPC) Anda atau memberi mereka kemampuan untuk mengunggah atau menghapus objek.

Saat Anda membuat URL presigned menggunakan AWS SDKs atau AWS Command Line Interface (AWS CLI), Anda mengaitkan URL dengan tindakan tertentu. Anda juga memberikan akses terbatas waktu ke URL yang telah ditetapkan sebelumnya dengan memilih waktu kedaluwarsa kustom yang bisa serendah 1 detik dan setinggi 7 hari. Saat Anda membagikan URL yang telah ditandatangani, individu di VPC dapat melakukan tindakan yang disertakan dalam URL seolah-olah mereka adalah pengguna yang memublikasikan sendiri. Ketika URL mencapai waktu kedaluwarsa, URL kedaluwarsa dan tidak lagi berfungsi.

Pembatasan kemampuan URL yang telah ditandatangani

Kemampuan URL yang telah ditandatangani dibatasi oleh izin pengguna yang membuatnya. Intinya, presigned URLs adalah token pembawa yang memberikan akses kepada mereka yang memilikinya. Oleh karena itu, kami menyarankan agar Anda melindunginya sebagaimana mestinya.

AWS Tanda Tangan Versi 4 (SiGv4)

Untuk menerapkan perilaku tertentu saat permintaan URL presigned diautentikasi menggunakan AWS Signature Version 4 (SigV4), Anda dapat menggunakan kunci kondisi dalam kebijakan bucket dan kebijakan jalur akses. Misalnya, Anda dapat membuat kebijakan bucket yang menggunakan `s3-outposts:signatureAge` kondisi untuk menolak permintaan URL Amazon S3 pada Outposts yang telah ditetapkan sebelumnya pada objek di `example-outpost-bucket` bucket jika tanda tangan berusia lebih dari 10 menit. Untuk menggunakan contoh ini, ganti *`user input placeholders`*dengan informasi Anda sendiri.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Deny a presigned URL request if the signature is more than 10
 minutes old",
            "Effect": "Deny",
            "Principal": {"AWS":"444455556666"},
            "Action": "s3-outposts:*",
```

```
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
            "Condition": {
                "NumericGreaterThan": {"s3-outposts:signatureAge": 600000},
                "StringEquals": {"s3-outposts:authType": "REST-QUERY-STRING"}
            }
        }
    ]
}
```

Untuk daftar kunci kondisi dan kebijakan contoh tambahan yang dapat Anda gunakan untuk menerapkan perilaku tertentu saat permintaan URL yang telah ditetapkan sebelumnya diautentikasi menggunakan Versi Tanda Tangan 4, lihat. AWS Kunci kebijakan khusus otentikasi Versi Tanda Tangan 4 (SiGv4)

Pembatasan jalur jaringan

Jika Anda ingin membatasi penggunaan presigned URLs dan semua S3 pada Outposts akses ke jalur jaringan tertentu, Anda dapat menulis kebijakan yang memerlukan jalur jaringan tertentu. Untuk menetapkan pembatasan pada prinsipal IAM yang membuat panggilan, Anda dapat menggunakan kebijakan berbasis identitas AWS Identity and Access Management (IAM) (misalnya, kebijakan pengguna, grup, atau peran). Untuk menetapkan batasan pada sumber daya S3 pada Outposts, Anda dapat menggunakan kebijakan berbasis sumber daya (misalnya, kebijakan bucket dan titik akses).

Pembatasan jalur jaringan pada pengguna utama IAM mengharuskan pengguna kredensial tersebut untuk membuat permintaan dari jaringan yang ditentukan. Pembatasan pada bucket atau titik akses mengharuskan semua permintaan ke sumber daya tersebut berasal dari jaringan tertentu. Pembatasan ini juga berlaku di luar skenario URL yang telah ditandatangani sebelumnya.

Kondisi global IAM yang Anda gunakan bergantung pada jenis titik akhir. Jika Anda menggunakan titik akhir publik untuk S3 di Outposts, gunakan `aws:SourceIp`. Jika Anda menggunakan titik akhir VPC untuk S3 di Outposts, gunakan `aws:SourceVpc` atau `aws:SourceVpce`.

Pernyataan kebijakan IAM berikut mengharuskan prinsipal untuk mengakses AWS hanya dari rentang jaringan yang ditentukan. Dengan pernyataan kebijakan ini, semua akses harus berasal dari rentang itu. Hal ini juga termasuk kasus seseorang yang menggunakan URL presigned untuk S3 di Outposts. Untuk menggunakan contoh ini, ganti *`user input placeholders`*dengan informasi Anda sendiri.

```
{
    "Sid": "NetworkRestrictionForIAMPrincipal",
    "Effect": "Deny",
    "Action": "*",
    "Resource": "*",
    "Condition": {
        "NotIpAddressIfExists": {"aws:SourceIp": "IP-address-range"},
        "BoolIfExists": {"aws:ViaAWSService": "false"}
    }
}
```

Untuk contoh kebijakan bucket yang menggunakan kunci kondisi `aws:SourceIP` AWS global untuk membatasi akses ke bucket S3 di Outposts ke rentang jaringan tertentu, lihat. Mengatur IAM dengan S3 di Outposts

Siapa yang dapat membuat URL yang telah ditandatangani

Siapa pun yang memiliki kredensial keamanan yang valid dapat membuat sebuah URL yang telah ditandatangani. Akan tetapi agar pengguna di VPC berhasil mengakses objek, URL presigned harus diciptakan oleh seseorang yang memiliki izin untuk melakukan operasi yang menjadi dasar dari URL presigned tersebut.

Anda dapat menggunakan kredensial berikut untuk membuat URL presigned:

- Profil instans IAM–Valid hingga 6 jam.
- AWS Security Token Service-Berlaku hingga 36 jam saat ditandatangani dengan kredensial permanen, seperti kredensial pengguna root Akun AWS atau pengguna IAM.
- Pengguna IAM - Berlaku hingga 7 hari saat Anda menggunakan AWS Signature Versi 4.

  Untuk menciptakan sebuah URL presigned yang berlaku hingga 7 hari, pertama-tama delegasikan kredensial pengguna IAM (kunci akses dan kunci rahasia) pada SDK yang Anda gunakan. Kemudian, buat URL presigned dengan menggunakan AWS Signature Version 4.

> **Note**
>
> - Jika Anda menciptakan sebuah URL presigned menggunakan token sementara, URL akan kedaluwarsa saat token kedaluwarsa, meskipun Anda membuat URL dengan waktu kedaluwarsa lebih lama.

- Karena akses URLs hibah yang telah ditetapkan sebelumnya ke bucket S3 on Outposts Anda kepada siapa pun yang memiliki URL, kami sarankan Anda melindunginya dengan tepat. Untuk informasi selengkapnya tentang melindungi presigned URLs, lihatPembatasan kemampuan URL yang telah ditandatangani.

Kapan S3 di Outposts memeriksa tanggal dan waktu kedaluwarsa dari URL yang telah ditandatangani?

Pada saat permintaan HTTP, S3 di Outposts memeriksa tanggal dan waktu kedaluwarsa dari URL yang ditandatangani. Misalnya, jika klien mulai mengunduh file besar segera sebelum waktu kedaluwarsa, pengunduhan berlanjut meskipun waktu kedaluwarsa berlalu selama pengunduhan. Akan tetapi, jika koneksi menurun dan klien mencoba memulai ulang unduhan setelah waktu kedaluwarsa berlalu, pengunduhan gagal.

Untuk informasi selengkapnya tentang menggunakan URL yang telah ditandatangani untuk berbagi atau unggah objek, lihat topik berikut ini.

Topik

- Berbagi objek dengan menggunakan presigned URLs
- Membuat sebuah URL presigned untuk mengunggah sebuah objek ke sebuah S3 di bucket Outposts

Berbagi objek dengan menggunakan presigned URLs

Untuk memberikan akses terbatas waktu ke objek yang disimpan secara lokal di Outpost tanpa memperbarui kebijakan bucket, Anda dapat menggunakan URL yang telah ditetapkan sebelumnya. Dengan presigned URLs, Anda sebagai pemilik bucket dapat berbagi objek dengan individu di virtual private cloud (VPC) Anda atau memberi mereka kemampuan untuk mengunggah atau menghapus objek.

Saat Anda membuat URL presigned menggunakan AWS SDKs atau AWS Command Line Interface (AWS CLI), Anda mengaitkan URL dengan tindakan tertentu. Anda juga memberikan akses terbatas waktu ke URL yang telah ditetapkan sebelumnya dengan memilih waktu kedaluwarsa kustom yang bisa serendah 1 detik dan setinggi 7 hari. Saat Anda membagikan URL yang telah ditandatangani, individu di VPC dapat melakukan tindakan yang disertakan dalam URL seolah-olah mereka

adalah pengguna yang memublikasikan sendiri. Ketika URL mencapai waktu kedaluwarsa, URL kedaluwarsa dan tidak lagi berfungsi.

Saat Anda membuat sebuah URL yang telah ditandatangani, Anda harus memberikan kredensial keamanan Anda, lalu menentukan berikut ini:

- Titik akses Amazon Resource Name (ARN) untuk Amazon S3 di bucket Outposts
- Kunci objek
- Metode HTTP (`GET`untuk mengunduh objek)
- Tanggal dan waktu kedaluwarsa

Sebuah URL presigned hanya berlaku selama durasi yang telah ditentukan. Artinya, Anda harus memulai tindakan yang diizinkan oleh URL sebelum tanggal dan waktu kedaluwarsa. Anda dapat menggunakan sebuah URL presigned berkali-kali, hingga tanggal dan waktu kedaluwarsa. Jika Anda membuat URL yang telah ditandatangani menggunakan token sementara, maka URL akan kedaluwarsa saat token kedaluwarsa, meskipun jika Anda membuat URL dengan waktu kedaluwarsa yang lebih lama.

Pengguna di virtual private cloud (VPC) yang memiliki akses ke URL presigned dapat mengakses objek. Misalnya, jika Anda memiliki video dalam bucket Anda dan bucket maupun objek tersebut bersifat pribadi, Anda dapat membagikan video dengan orang lain dengan membuat URL presigned. Karena akses URLs hibah yang telah ditetapkan sebelumnya ke bucket S3 on Outposts Anda kepada siapa pun yang memiliki URL, kami sarankan Anda melindunginya dengan tepat. URLs Untuk detail selengkapnya tentang melindungi presigned URLs, lihatPembatasan kemampuan URL yang telah ditandatangani.

Siapa pun yang memiliki kredensial keamanan yang valid dapat membuat sebuah URL yang telah ditandatangani. Akan tetapi, URL presigned harus dibuat oleh seseorang yang memiliki izin untuk melakukan operasi yang menjadi dasar URL presigned. Untuk informasi selengkapnya, lihat Siapa yang dapat membuat URL yang telah ditandatangani.

Anda dapat membuat URL yang telah ditetapkan sebelumnya untuk berbagi objek di bucket S3 di Outposts dengan menggunakan dan file. AWS SDKs AWS CLI Untuk informasi selengkapnya, lihat contoh berikut ini.

Menggunakan AWS SDKs

Anda dapat menggunakan AWS SDKs untuk menghasilkan URL presigned yang dapat Anda berikan kepada orang lain sehingga mereka dapat mengambil objek.

> **Note**
>
> Saat Anda menggunakan AWS SDKs untuk membuat URL yang telah ditetapkan sebelumnya, waktu kedaluwarsa maksimum untuk URL yang telah ditetapkan sebelumnya adalah 7 hari dari waktu pembuatan.

Java

Example

Contoh berikut menampilkan pembuatan URL presigned yang dapat Anda berikan kepada orang lain sehingga mereka dapat mengambil objek dari bucket S3 di Outposts. Untuk informasi selengkapnya, lihat Menggunakan presigned URLs untuk S3 di Outposts. Untuk menggunakan contoh ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.HttpMethod;
import com.amazonaws.SdkClientException;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.GeneratePresignedUrlRequest;

import java.io.IOException;
import java.net.URL;
import java.time.Instant;

public class GeneratePresignedURL {

    public static void main(String[] args) throws IOException {
        Regions clientRegion = Regions.DEFAULT_REGION;
        String accessPointArn = "*** access point ARN ***";
        String objectKey = "*** object key ***";

        try {
```

```
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .withRegion(clientRegion)
                    .withCredentials(new ProfileCredentialsProvider())
                    .build();

            // Set the presigned URL to expire after one hour.
            java.util.Date expiration = new java.util.Date();
            long expTimeMillis = Instant.now().toEpochMilli();
            expTimeMillis += 1000 * 60 * 60;
            expiration.setTime(expTimeMillis);

            // Generate the presigned URL.
            System.out.println("Generating pre-signed URL.");
            GeneratePresignedUrlRequest generatePresignedUrlRequest =
                    new GeneratePresignedUrlRequest(accessPointArn, objectKey)
                            .withMethod(HttpMethod.GET)
                            .withExpiration(expiration);
            URL url = s3Client.generatePresignedUrl(generatePresignedUrlRequest);

            System.out.println("Pre-Signed URL: " + url.toString());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't
 process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

.NET

Example

Contoh berikut menampilkan pembuatan URL presigned yang dapat Anda berikan kepada orang lain sehingga mereka dapat mengambil objek dari bucket S3 di Outposts. Untuk informasi selengkapnya, lihat Menggunakan presigned URLs untuk S3 di Outposts. Untuk menggunakan contoh ini, ganti *user input placeholders*dengan informasi Anda sendiri.

```
using Amazon;
```

```
using Amazon.S3;
using Amazon.S3.Model;
using System;

namespace Amazon.DocSamples.S3
{
    class GenPresignedURLTest
    {
        private const string accessPointArn = "*** access point ARN ***";
        private const string objectKey = "*** object key ***";
        // Specify how long the presigned URL lasts, in hours.
        private const double timeoutDuration = 12;
        // Specify your bucket Region (an example Region is shown).
        private static readonly RegionEndpoint bucketRegion =
 RegionEndpoint.USWest2;
        private static IAmazonS3 s3Client;

        public static void Main()
        {
            s3Client = new AmazonS3Client(bucketRegion);
            string urlString = GeneratePreSignedURL(timeoutDuration);
        }
        static string GeneratePreSignedURL(double duration)
        {
            string urlString = "";
            try
            {
                GetPreSignedUrlRequest request1 = new GetPreSignedUrlRequest
                {
                    BucketName = accessPointArn,
                    Key = objectKey,
                    Expires = DateTime.UtcNow.AddHours(duration)
                };
                urlString = s3Client.GetPreSignedURL(request1);
            }
            catch (AmazonS3Exception e)
            {
                Console.WriteLine("Error encountered on server. Message:'{0}' when
 writing an object", e.Message);
            }
            catch (Exception e)
            {
                Console.WriteLine("Unknown encountered on server. Message:'{0}' when
 writing an object", e.Message);
```

```
            }
            return urlString;
        }
    }
}
```

Python

Contoh berikut menampilkan pembuatan URL presigned untuk berbagi objek dengan menggunakan SDK untuk Python (Boto3). Sebagai contoh, gunakan klien Boto3 dan `generate_presigned_url` fungsi untuk menghasilkan URL presigned yang memungkinkan Anda untuk `GET` objek.

```
import boto3
    url = boto3.client('s3').generate_presigned_url(
    ClientMethod='get_object',
    Params={'Bucket': 'ACCESS_POINT_ARN', 'Key': 'OBJECT_KEY'},
    ExpiresIn=3600)
```

Untuk informasi lebih lanjut tentang penggunaan SDK untuk Python (Boto3) untuk menghasilkan URL yang telah ditandatangani sebelumnya, lihat Python di Referensi API AWS SDK for Python (Boto) .

Menggunakan AWS CLI

AWS CLI Perintah contoh berikut menghasilkan URL presigned untuk bucket S3 pada Outposts. Untuk menggunakan contoh ini, ganti *`user input placeholders`*dengan informasi Anda sendiri.

> **Note**
>
> Saat Anda menggunakan AWS CLI untuk membuat URL yang telah ditetapkan sebelumnya, waktu kedaluwarsa maksimum untuk URL yang telah ditetapkan sebelumnya adalah 7 hari dari waktu pembuatan.

```
aws s3 presign s3://arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/example-outpost-access-
point/mydoc.txt --expires-in 604800
```

Untuk informasi selengkapnya, lihat presign dalam Referensi AWS CLI Perintah.

Membuat sebuah URL presigned untuk mengunggah sebuah objek ke sebuah S3 di bucket Outposts

Untuk memberikan akses terbatas waktu ke objek yang disimpan secara lokal di Outpost tanpa memperbarui kebijakan bucket, Anda dapat menggunakan URL yang telah ditetapkan sebelumnya. Dengan presigned URLs, Anda sebagai pemilik bucket dapat berbagi objek dengan individu di virtual private cloud (VPC) Anda atau memberi mereka kemampuan untuk mengunggah atau menghapus objek.

Saat Anda membuat URL presigned menggunakan AWS SDKs atau AWS Command Line Interface (AWS CLI), Anda mengaitkan URL dengan tindakan tertentu. Anda juga memberikan akses terbatas waktu ke URL yang telah ditetapkan sebelumnya dengan memilih waktu kedaluwarsa kustom yang bisa serendah 1 detik dan setinggi 7 hari. Saat Anda membagikan URL yang telah ditandatangani, individu di VPC dapat melakukan tindakan yang disertakan dalam URL seolah-olah mereka adalah pengguna yang memublikasikan sendiri. Ketika URL mencapai waktu kedaluwarsa, URL kedaluwarsa dan tidak lagi berfungsi.

Saat Anda membuat sebuah URL yang telah ditandatangani, Anda harus memberikan kredensial keamanan Anda, lalu menentukan berikut ini:

- Titik akses Amazon Resource Name (ARN) untuk Amazon S3 di bucket Outposts
- Kunci objek
- Metode HTTP (`PUT`untuk mengunggah objek)
- Tanggal dan waktu kedaluwarsa

Sebuah URL presigned hanya berlaku selama durasi yang telah ditentukan. Artinya, Anda harus memulai tindakan yang diizinkan oleh URL sebelum tanggal dan waktu kedaluwarsa. Anda dapat menggunakan sebuah URL presigned berkali-kali, hingga tanggal dan waktu kedaluwarsa. Jika Anda membuat URL yang telah ditandatangani menggunakan token sementara, maka URL akan kedaluwarsa saat token kedaluwarsa, meskipun jika Anda membuat URL dengan waktu kedaluwarsa yang lebih lama.

Jika tindakan yang diizinkan oleh sebuah URL presigned terdiri dari beberapa langkah, seperti unggahan multipart, Anda harus memulai semua langkah sebelum waktu kedaluwarsa. Jika S3 di Outposts mencoba memulai langkah dengan URL yang kedaluwarsa, Anda menerima kesalahan.

Pengguna di virtual private cloud (VPC) yang memiliki akses ke URL presigned dapat mengunggah objek. Misalnya, pengguna di VPC yang memiliki akses ke URL yang telah ditetapkan sebelumnya

dapat mengunggah objek ke bucket Anda. Karena URLs akses yang diberikan sebelumnya ke bucket S3 on Outposts Anda kepada pengguna mana pun di VPC yang memiliki akses ke URL yang telah ditetapkan sebelumnya, sebaiknya Anda melindunginya dengan tepat. URLs Untuk detail selengkapnya tentang melindungi presigned URLs, lihatPembatasan kemampuan URL yang telah ditandatangani.

Siapa pun yang memiliki kredensial keamanan yang valid dapat membuat sebuah URL yang telah ditandatangani. Akan tetapi, URL presigned harus dibuat oleh seseorang yang memiliki izin untuk melakukan operasi yang menjadi dasar URL presigned. Untuk informasi selengkapnya, lihat Siapa yang dapat membuat URL yang telah ditandatangani.

Menggunakan AWS SDKs untuk menghasilkan URL presigned untuk S3 pada operasi objek Outposts

Java

SDK untuk Java 2.x

Contoh ini menunjukkan cara menghasilkan sebuah URL presigned yang dapat digunakan untuk mengunggah sebuah objek ke sebuah bucket S3 di Outposts selama waktu yang terbatas. Untuk informasi selengkapnya, lihat Menggunakan presigned URLs untuk S3 di Outposts.

```
   public static void signBucket(S3Presigner presigner, String
outpostAccessPointArn, String keyName) {

       try {
           PutObjectRequest objectRequest = PutObjectRequest.builder()
                   .bucket(accessPointArn)
                   .key(keyName)
                   .contentType("text/plain")
                   .build();

           PutObjectPresignRequest presignRequest =
PutObjectPresignRequest.builder()
                   .signatureDuration(Duration.ofMinutes(10))
                   .putObjectRequest(objectRequest)
                   .build();

           PresignedPutObjectRequest presignedRequest =
presigner.presignPutObject(presignRequest);
```

```
            String myURL = presignedRequest.url().toString();
            System.out.println("Presigned URL to upload a file to: " +myURL);
            System.out.println("Which HTTP method must be used when uploading a
file: " +
                    presignedRequest.httpRequest().method());

            // Upload content to the S3 on Outposts bucket by using this URL.
            URL url = presignedRequest.url();

            // Create the connection and use it to upload the new object by using
the presigned URL.
            HttpURLConnection connection = (HttpURLConnection)
url.openConnection();
            connection.setDoOutput(true);
            connection.setRequestProperty("Content-Type","text/plain");
            connection.setRequestMethod("PUT");
            OutputStreamWriter out = new
OutputStreamWriter(connection.getOutputStream());
            out.write("This text was uploaded as an object by using a presigned
URL.");
            out.close();

            connection.getResponseCode();
            System.out.println("HTTP response code is " +
connection.getResponseCode());

        } catch (S3Exception e) {
            e.getStackTrace();
        } catch (IOException e) {
            e.getStackTrace();
        }
    }
```

Python

SDK untuk Python (Boto3)

Contoh ini menunjukkan cara membuat URL presigned yang dapat melakukan tindakan S3 on Outposts untuk waktu yang terbatas. Untuk informasi selengkapnya, lihat Menggunakan

presigned URLs untuk S3 di Outposts. Untuk membuat permintaan dengan URL, gunakan `Requests` paket.

```
import argparse
import logging
import boto3
from botocore.exceptions import ClientError
import requests

logger = logging.getLogger(__name__)


def generate_presigned_url(s3_client, client_method, method_parameters,
 expires_in):
    """
    Generate a presigned S3 on Outposts URL that can be used to perform an
 action.

    :param s3_client: A Boto3 Amazon S3 client.
    :param client_method: The name of the client method that the URL performs.
    :param method_parameters: The parameters of the specified client method.
    :param expires_in: The number of seconds that the presigned URL is valid for.
    :return: The presigned URL.
    """
    try:
        url = s3_client.generate_presigned_url(
            ClientMethod=client_method,
            Params=method_parameters,
            ExpiresIn=expires_in
        )
        logger.info("Got presigned URL: %s", url)
    except ClientError:
        logger.exception(
            "Couldn't get a presigned URL for client method '%s'.",
 client_method)
        raise
    return url


def usage_demo():
    logging.basicConfig(level=logging.INFO, format='%(levelname)s: %(message)s')

    print('-'*88)
```

```
    print("Welcome to the Amazon S3 on Outposts presigned URL demo.")
    print('-'*88)

    parser = argparse.ArgumentParser()
    parser.add_argument('accessPointArn', help="The name of the S3 on Outposts
access point ARN.")
    parser.add_argument(
        'key', help="For a GET operation, the key of the object in S3 on
Outposts. For a "
                    "PUT operation, the name of a file to upload.")
    parser.add_argument(
        'action', choices=('get', 'put'), help="The action to perform.")
    args = parser.parse_args()

    s3_client = boto3.client('s3')
    client_action = 'get_object' if args.action == 'get' else 'put_object'
    url = generate_presigned_url(
        s3_client, client_action, {'Bucket': args.accessPointArn, 'Key':
args.key}, 1000)

    print("Using the Requests package to send a request to the URL.")
    response = None
    if args.action == 'get':
        response = requests.get(url)
    elif args.action == 'put':
        print("Putting data to the URL.")
        try:
            with open(args.key, 'r') as object_file:
                object_text = object_file.read()
            response = requests.put(url, data=object_text)
        except FileNotFoundError:
            print(f"Couldn't find {args.key}. For a PUT operation, the key must
be the "
                  f"name of a file that exists on your computer.")

    if response is not None:
        print("Got response:")
        print(f"Status: {response.status_code}")
        print(response.text)

    print('-'*88)


if __name__ == '__main__':
```

```
    usage_demo()
```

Amazon S3 di Outposts dengan Amazon EMR lokal di Outposts

Amazon EMR adalah platform cluster terkelola yang menyederhanakan menjalankan kerangka kerja data besar, seperti Apache Hadoop and Apache Spark, AWS untuk memproses dan menganalisis sejumlah besar data. Dengan menggunakan kerangka kerja ini dan proyek sumber terbuka terkait, Anda dapat memproses data untuk tujuan analitik dan beban kerja intelijen bisnis. Amazon EMR juga membantu Anda mengubah dan memindahkan sejumlah besar data ke dalam dan keluar dari penyimpanan AWS data dan database lainnya, serta mendukung Amazon S3 di Outposts. Untuk informasi selengkapnya tentang Amazon EMR, lihat Amazon EMR di Outposts di Panduan Manajemen EMR Amazon.

Untuk Amazon S3 di Outposts, Amazon EMR mulai mendukung Apache Hadoop Konektor S3A dalam versi 7.0.0. Versi Amazon EMR sebelumnya tidak mendukung S3 lokal di Outposts, dan EMR File System (EMRFS) tidak didukung.

Aplikasi-aplikasi yang didukung

Amazon EMR dengan Amazon S3 di Outposts mendukung aplikasi berikut:

- Hadoop
- Spark
- Hue
- Hive
- Sqoop
- Pig
- Hudi
- Flink

Untuk informasi selengkapnya, lihat Panduan Rilis Amazon EMR.

Membuat dan mengonfigurasi bucket Amazon S3 di Outposts

Amazon EMR menggunakan AWS SDK untuk Java dengan Amazon S3 di Outposts untuk menyimpan data input dan data output. File log EMR Amazon Anda disimpan di lokasi Amazon S3

Regional yang Anda pilih dan tidak disimpan secara lokal di Outpost. Untuk informasi selengkapnya, lihat log EMR Amazon di Panduan Manajemen EMR Amazon.

Agar sesuai dengan persyaratan Amazon S3 dan DNS, bucket S3 on Outposts memiliki batasan dan batasan penamaan. Untuk informasi selengkapnya, lihat Membuat bucket S3 di Outposts.

Dengan Amazon EMR versi 7.0.0 dan yang lebih baru, Anda dapat menggunakan Amazon EMR dengan S3 di Outposts dan sistem file S3A.

Prasyarat

Izin S3 on Outposts — Saat Anda membuat profil instans EMR Amazon, peran Anda harus berisi namespace (IAM) untuk S3 AWS Identity and Access Management di Outposts. S3 di Outposts memiliki namespace sendiri,. `s3-outposts*` Untuk contoh kebijakan yang menggunakan namespace ini, lihat. Mengatur IAM dengan S3 di Outposts

Konektor S3A - Untuk mengonfigurasi kluster EMR Anda untuk mengakses data dari bucket Amazon S3 di Outposts, Anda harus menggunakan Apache Hadoop Konektor S3A. Untuk menggunakan konektor, pastikan semua S3 Anda URIs menggunakan `s3a` skema. Jika tidak, Anda dapat mengonfigurasi implementasi sistem file yang Anda gunakan untuk cluster EMR Anda sehingga S3 Anda URIs bekerja dengan konektor S3A.

Untuk mengonfigurasi implementasi sistem file agar berfungsi dengan konektor S3A, Anda menggunakan properti `fs.`*`file_scheme`*`.impl` dan `fs.AbstractFileSystem.`*`file_scheme`*`.impl` konfigurasi untuk cluster EMR Anda, yang *`file_scheme`* sesuai dengan jenis URIs S3 yang Anda miliki. Untuk menggunakan contoh berikut, ganti *`user input placeholders`* dengan informasi Anda sendiri. Misalnya, untuk mengubah implementasi sistem file untuk S3 URIs yang menggunakan `s3` skema, tentukan properti konfigurasi cluster berikut:

```
[
  {
"Classification": "core-site",
    "Properties": {
    "fs.s3.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
    "fs.AbstractFileSystem.s3.impl": "org.apache.hadoop.fs.s3a.S3A"
    }
  }
]
```

Untuk menggunakan S3A, atur properti `fs.file_scheme.impl` konfigurasi ke`org.apache.hadoop.fs.s3a.S3AFileSystem`, dan atur `fs.AbstractFileSystem.file_scheme.impl` properti ke. `org.apache.hadoop.fs.s3a.S3A`

Misalnya, jika Anda mengakses jalur`s3a://bucket/...`, atur `fs.s3a.impl` properti ke`org.apache.hadoop.fs.s3a.S3AFileSystem`, dan setel `fs.AbstractFileSystem.s3a.impl` properti ke`org.apache.hadoop.fs.s3a.S3A`.

Memulai menggunakan Amazon EMR dengan Amazon S3 di Outposts

Topik berikut menjelaskan cara memulai menggunakan Amazon EMR dengan Amazon S3 di Outposts.

Topik

- Membuat kebijakan izin
- Buat dan konfigurasikan cluster Anda
- Ikhtisar konfigurasi
- Pertimbangan

Membuat kebijakan izin

Sebelum Anda dapat membuat kluster EMR yang menggunakan Amazon S3 di Outposts, Anda harus membuat kebijakan IAM untuk melampirkan ke profil instans EC2 Amazon untuk cluster. Kebijakan harus memiliki izin untuk mengakses jalur akses S3 di Outposts Nama Sumber Daya Amazon (ARN). Untuk informasi selengkapnya tentang membuat kebijakan IAM untuk S3 di Outposts, lihat. Mengatur IAM dengan S3 di Outposts

Kebijakan contoh berikut menunjukkan cara memberikan izin yang diperlukan. Setelah membuat kebijakan, lampirkan kebijakan ke peran profil instance yang Anda gunakan untuk membuat klaster EMR, seperti yang dijelaskan di bagian inithe section called “Buat dan konfigurasikan cluster Anda”. Untuk menggunakan contoh ini, ganti *user input placeholders* dengan informasi Anda sendiri.

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

```
            "Resource": "arn:aws:s3-outposts:us-
west-2:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/access-point-name",
            "Action": [
                "s3-outposts:*"
            ]
        }
    ]

}
```

Buat dan konfigurasikan cluster Anda

Untuk membuat cluster yang menjalankan Spark dengan S3 di Outposts, selesaikan langkah-langkah berikut di konsol.

Untuk membuat cluster yang berjalan Spark dengan S3 di Outposts

1. Buka konsol Amazon EMR. di https://console.aws.amazon.com/elasticmapreduce/.
2. Pada panel navigasi sebelah kiri, pilih Klaster.
3. Pilih Buat klaster.
4. Untuk rilis Amazon EMR, pilih emr-7.0.0atau nanti.
5. Untuk bundel Aplikasi, pilih Spark interaktif. Kemudian pilih aplikasi lain yang didukung yang ingin Anda sertakan dalam cluster Anda.
6. Untuk mengaktifkan Amazon S3 di Outposts, masukkan pengaturan konfigurasi Anda.

   Contoh pengaturan konfigurasi

   Untuk menggunakan pengaturan konfigurasi sampel berikut, ganti *user input placeholders* dengan informasi Anda sendiri.

   ```
   [
    {
      "Classification": "core-site",
      "Properties": {
        "fs.s3a.bucket.DOC-EXAMPLE-BUCKET.accesspoint.arn": "arn:aws:s3-outposts:us-
   west-2:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/access-point-name"
        "fs.s3a.committer.name": "magic",
        "fs.s3a.select.enabled": "false"
       }
   ```

```
  },
  {
    "Classification": "hadoop-env",
    "Configurations": [
      {
        "Classification": "export",
        "Properties": {
          "JAVA_HOME": "/usr/lib/jvm/java-11-amazon-corretto.x86_64"
          }
       }
     ],
     "Properties": {}
   },
   {
     "Classification": "spark-env",
     "Configurations": [
       {
         "Classification": "export",
         "Properties": {
           "JAVA_HOME": "/usr/lib/jvm/java-11-amazon-corretto.x86_64"
         }
       }
      ],
      "Properties": {}
     },
     {
      "Classification": "spark-defaults",
      "Properties": {
        "spark.executorEnv.JAVA_HOME": "/usr/lib/jvm/java-11-amazon-
corretto.x86_64",
        "spark.sql.sources.fastS3PartitionDiscovery.enabled": "false"
      }
     }
  ]
```

7. Di bagian Networking, pilih virtual private cloud (VPC) dan subnet yang ada di rak Anda. AWS Outposts Untuk informasi selengkapnya tentang Amazon EMR di Outposts, lihat kluster EMR di Panduan Manajemen EMR Amazon. AWS Outposts
8. Di bagian profil EC2 instans untuk Amazon EMR, pilih peran IAM yang memiliki kebijakan izin yang telah dilampirkan sebelumnya.
9. Konfigurasikan setelan cluster Anda yang tersisa, lalu pilih Create cluster.

Ikhtisar konfigurasi

Tabel berikut menjelaskan konfigurasi S3A dan nilai yang akan ditentukan parameternya saat Anda menyiapkan cluster yang menggunakan S3 di Outposts dengan Amazon EMR.

Parameter	Nilai default	Nilai yang diperlukan untuk S3 di Outposts	Penjelasan
`fs.s3a.aws.credentials.provider`	Jika tidak ditentukan, S3A akan mencari S3 di bucket Region dengan nama bucket Outposts.	Titik akses ARN dari ember S3 on Outposts	Amazon S3 di Outposts mendukung titik akses khusus virtual private cloud (VPC) sebagai satu-satunya cara untuk mengakses bucket Outposts.
`fs.s3a.committer.name`	`file`	`magic`	Magic committer adalah satu-satunya committer yang didukung untuk S3 di Outposts.
`fs.s3a.select.enabled`	`TRUE`	`FALSE`	S3 Select tidak didukung di Outposts.
`JAVA_HOME`	`/usr/lib/jvm/java-8`	`/usr/lib/jvm/java-11-amazon-corretto.x86_64`	S3 di Outposts di S3A membutuhkan Java versi 11.

Tabel berikut menjelaskan Spark konfigurasi dan nilai yang akan ditentukan parameternya saat Anda menyiapkan cluster yang menggunakan S3 di Outposts dengan Amazon EMR.

Parameter	Nilai default	Nilai yang diperlukan untuk S3 di Outposts	Penjelasan
`spark.sql.sources.fastS3PartitionDiscovery.enabled`	`TRUE`	`FALSE`	S3 di Outposts tidak mendukung partisi cepat.
`spark.executorEnv.JAVA_HOME`	`/usr/lib/jvm/java-8`	`/usr/lib/jvm/java-11-amazon-corretto.x86_64`	S3 di Outposts pada S3A membutuhkan Java versi 11.

Pertimbangan

Pertimbangkan hal berikut saat Anda mengintegrasikan Amazon EMR dengan S3 pada bucket Outposts:

- Amazon S3 di Outposts didukung dengan Amazon EMR versi 7.0.0 dan yang lebih baru.
- Konektor S3A diperlukan untuk menggunakan S3 di Outposts dengan Amazon EMR. Hanya S3A yang memiliki fitur yang diperlukan untuk berinteraksi dengan S3 pada bucket Outposts. Untuk informasi penyiapan konektor S3A, lihat Prasyarat.
- Amazon S3 di Outposts hanya mendukung enkripsi sisi server dengan kunci terkelola Amazon S3 (SSE-S3) dengan Amazon EMR. Untuk informasi selengkapnya, lihat the section called “Enkripsi data”.
- Amazon S3 di Outposts tidak mendukung penulisan dengan S3A. FileOutputCommitter Menulis dengan S3A FileOutputCommitter pada S3 pada bucket Outposts menghasilkan kesalahan berikut: InvalidStorageClass: Kelas penyimpanan yang Anda tentukan tidak valid.
- Amazon S3 di Outposts tidak didukung dengan Amazon EMR Tanpa Server atau Amazon EMR di EKS.
- Log EMR Amazon disimpan di lokasi Amazon S3 Regional yang Anda pilih, dan tidak disimpan secara lokal di bucket S3 on Outposts.

Caching otorisasi dan otentikasi

S3 di Outposts dengan aman menyimpan data otentikasi dan otorisasi secara lokal di rak Outposts. Cache menghapus perjalanan pulang pergi ke induk Wilayah AWS untuk setiap permintaan. Ini menghilangkan variabilitas yang diperkenalkan oleh perjalanan pulang pergi jaringan. Dengan cache otentikasi dan otorisasi di S3 di Outposts, Anda mendapatkan latensi konsisten yang independen dari latensi koneksi antara Outposts dan Outposts. Wilayah AWS

Saat Anda membuat permintaan API S3 on Outposts, data autentikasi dan otorisasi di-cache dengan aman. Data yang di-cache kemudian digunakan untuk mengautentikasi permintaan API objek S3 berikutnya. S3 di Outposts hanya menyimpan data otentikasi dan otorisasi cache saat permintaan ditandatangani menggunakan Signature Version 4A (Sigv4a). Cache disimpan secara lokal di Outposts dalam layanan S3 on Outposts. Ini menyegarkan secara asinkron saat Anda membuat permintaan API S3. Cache dienkripsi, dan tidak ada kunci kriptografi plaintext yang disimpan di Outposts.

Cache berlaku hingga 10 menit ketika Outpost terhubung ke file. Wilayah AWS Ini disegarkan secara asinkron saat Anda membuat permintaan S3 pada Outposts API, untuk memastikan bahwa kebijakan terbaru digunakan. Jika Outpost terputus dari Wilayah AWS, cache akan berlaku hingga 12 jam.

Mengkonfigurasi cache otorisasi dan otentikasi

S3 di Outposts secara otomatis menyimpan data otentikasi dan otorisasi untuk permintaan yang ditandatangani dengan algoritma Sigv4a. Untuk informasi selengkapnya, lihat Menandatangani permintaan AWS API di Panduan AWS Identity and Access Management Pengguna. Algoritma Sigv4a tersedia dalam versi terbaru dari file. AWS SDKs Anda dapat memperolehnya melalui ketergantungan pada pustaka AWS Common Runtime (CRT).

Anda perlu menggunakan AWS SDK versi terbaru dan menginstal CRT versi terbaru. Misalnya, Anda dapat menjalankan `pip install awscrt` untuk mendapatkan CRT versi terbaru dengan Boto3.

S3 di Outposts tidak menyimpan data otentikasi dan otorisasi cache untuk permintaan yang ditandatangani dengan algoritma SiGv4.

Memvalidasi penandatanganan Sigv4a

Anda dapat menggunakan AWS CloudTrail untuk memvalidasi bahwa permintaan telah ditandatangani dengan Sigv4a. Untuk informasi lebih lanjut tentang pengaturan CloudTrail untuk S3 di Outposts, lihat. Memantau S3 di AWS CloudTrail Outposts dengan log

Setelah Anda mengonfigurasi CloudTrail, Anda dapat memverifikasi bagaimana permintaan ditandatangani di `SignatureVersion` bidang CloudTrail log. Permintaan yang ditandatangani dengan Sigv4a akan disetel ke`SignatureVersion`. `AWS 4-ECDSA-P256-SHA256` Permintaan yang ditandatangani dengan SiGv4 akan `SignatureVersion` disetel ke. `AWS 4-HMAC-SHA256`

Keamanan di S3 di Outposts

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 berjalan Layanan AWS di dalamnya AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program AWS Kepatuhan Program AWS Kepatuhan .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh Layanan AWS yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, termasuk sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini akan membantu Anda dalam memahami cara menerapkan model tanggung jawab bersama saat Anda menggunakan S3 di Outposts. Topik berikut menunjukkan kepada Anda cara mengonfigurasi S3 di Outposts untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga belajar cara menggunakan Layanan AWS yang lain yang membantu Anda memantau dan mengamankan sumber daya S3 Anda di Outposts.

Topik

Mengatur IAM dengan S3 di Outposts

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengendalikan siapa

saja yang dapat diautentikasi (masuk) dan diizinkan (memiliki izin) untuk menggunakan sumber daya Amazon S3 di Outposts. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan. Secara default, pengguna tidak memiliki izin untuk sumber daya dan operasi S3 di Outposts. Untuk memberikan izin akses bagi sumber daya dan operasi API S3 di Outposts, Anda dapat menggunakan IAM untuk membuat pengguna, grup, atau peran dan melampirkan izin.

Untuk memberikan akses, menambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam 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.

Selain kebijakan berbasis identitas IAM, S3 di Outposts mendukung kebijakan bucket dan titik akses. Kebijakan bucket dan titik akses adalah kebijakan berbasis sumber daya yang dilampirkan ke sumber daya S3 di Outposts.

- Kebijakan bucket dilampirkan pada bucket dan mengizinkan atau menolak permintaan ke bucket serta objek di dalamnya berdasarkan elemen dalam kebijakan.
- Sebaliknya, kebijakan titik akses dilampirkan ke titik akses dan memungkinkan atau menolak permintaan ke titik akses.

Kebijakan titik akses bekerja dengan kebijakan bucket yang dilampirkan pada bucket yang mendasari S3 di Outposts. Agar aplikasi atau pengguna dapat mengakses objek di bucket S3 di Outposts melalui titik akses S3 di Outposts, kebijakan titik akses dan kebijakan bucket harus mengizinkan permintaan tersebut.

Pembatasan yang Anda sertakan dalam kebijakan titik akses hanya berlaku untuk permintaan yang dibuat melalui titik akses tersebut. Misalnya, jika titik akses dilampirkan ke bucket, Anda tidak dapat menggunakan kebijakan titik akses untuk mengizinkan atau menolak permintaan yang dibuat langsung ke bucket. Namun, pembatasan yang Anda terapkan pada kebijakan bucket dapat mengizinkan atau menolak permintaan yang dibuat langsung ke bucket atau melalui titik akses.

Dalam kebijakan IAM atau kebijakan berbasis sumber daya, Anda menentukan tindakan S3 di Outposts mana yang diizinkan atau ditolak. Tindakan S3 di Outposts sesuai dengan operasi API S3 di Outposts tertentu. Tindakan S3 di Outposts menggunakan prefiks namespace `s3-outposts:`. Permintaan yang dibuat ke S3 on Outposts mengontrol API dalam dan permintaan Wilayah AWS yang dibuat ke titik akhir API objek di Outpost diautentikasi dengan menggunakan IAM dan diotorisasi terhadap awalan namespace. `s3-outposts:` Untuk bekerja dengan S3 di Outposts, konfigurasikan pengguna IAM Anda dan izinkan mereka terhadap namespace IAM `s3-outposts:`.

Untuk informasi lebih lanjut, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon S3 di Outposts di Referensi Otorisasi Layanan.

> **Note**
>
> - Daftar kontrol akses (ACLs) tidak didukung oleh S3 di Outposts.
> - S3 di Outposts adalah default bagi pemilik bucket sebagai pemilik objek, untuk membantu memastikan bahwa pemilik bucket tidak dapat dicegah untuk mengakses atau menghapus objek.
> - S3 di Outposts selalu mengaktifkan Blokir Akses Publik S3 untuk membantu memastikan objek tidak pernah memiliki akses publik.

Untuk informasi tentang pengaturan IAM untuk S3 di Outposts, lihat topik berikut.

Topik

- Pengguna utama kebijakan S3 di Outposts
- Sumber daya ARNs untuk S3 di Outposts
- Contoh kebijakan untuk S3 di Outposts
- Izin untuk titik akhir S3 di Outposts
- Peran yang ditautkan dengan layanan untuk S3 di Outposts

Pengguna utama kebijakan S3 di Outposts

Saat membuat kebijakan berbasis sumber daya untuk memberikan akses ke bucket S3 di Outposts, Anda harus menggunakan elemen `Principal` tersebut untuk menentukan orang atau aplikasi yang dapat membuat permintaan tindakan atau operasi pada sumber daya tersebut. Untuk kebijakan S3 di Outposts, Anda dapat menggunakan salah satu pengguna utama berikut:

- Sebuah Akun AWS
- Pengguna IAM
- Peran IAM
- Semua pengguna utama, dengan menentukan karakter wildcard (*) dalam kebijakan yang menggunakan elemen `Condition` untuk membatasi akses ke rentang IP tertentu

> **Important**
>
> Anda tidak dapat menulis kebijakan untuk bucket S3 di Outposts yang menggunakan karakter wildcard (`*`) dalam elemen `Principal` kecuali jika kebijakan tersebut juga menyertakan `Condition` yang membatasi akses ke rentang alamat IP tertentu. Pembatasan ini membantu memastikan tidak ada akses publik ke bucket S3 di Outposts Anda. Sebagai contoh, lihat Contoh kebijakan untuk S3 di Outposts.

Untuk informasi selengkapnya tentang elemen `Principal`, lihat elemen kebijakan JSON AWS : Pengguna utama dalam Panduan Pengguna IAM.

Sumber daya ARNs untuk S3 di Outposts

Amazon Resource Names (ARNs) untuk S3 di Outposts berisi ID Outpost selain Wilayah AWS tempat Outpost berada, ID, Akun AWS dan nama resource. Untuk mengakses dan melakukan tindakan pada bucket dan objek Outposts Anda, Anda harus menggunakan salah satu format ARN yang ditunjukkan pada tabel berikut.

*`partition`*Nilai dalam ARN mengacu pada sekelompok. Wilayah AWS Masing-masing Akun AWS dicakup ke satu partisi. Berikut ini adalah partisi yang didukung:

- `aws` – Wilayah AWS
- `aws-us-gov`— AWS GovCloud (US) Daerah

Tabel berikut menunjukkan S3 pada Outposts format ARN.

ARN Amazon S3 di Outposts	Format ARN	Contoh
ARN Bucket	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id* /bucket/*bucket_name*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904* /bucket/*amzn-s3-demo-bucket1*
Titik Akses ARN	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id* /accesspoint/ *accesspoint_name*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904* /accesspoint/ *access-point-name*
ARN objek	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id* /bucket/*bucket_name* /object/*object_key*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904* /bucket/*amzn-s3-demo-bucket1* */object/myobject*
ARN objek titik akses S3 di Outposts (digunakan dalam kebijakan)	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id* /accesspoint/ *accesspoint_name* /object/*object_key*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904* /accesspoint/ *access-point-name/object/myobject*

ARN Amazon S3 di Outposts	Format ARN	Contoh
ARN S3 di Outposts	`arn:`*`partition`*` :s3-outposts: `*`region`*`: `*`account_id`*` :outpost/`*`outpost_id`*	`arn:`*`aws`*`:s3-outposts: `*`us-west-2`*` :`*`123456789012`*` :outpost/ `*`op-01ac5d28a6a232904`*

Contoh kebijakan untuk S3 di Outposts

Example : S3 tentang kebijakan bucket Outposts dengan kepala sekolah Akun AWS

Kebijakan bucket berikut menggunakan Akun AWS prinsipal untuk memberikan akses ke bucket S3 di Outposts. Untuk menggunakan kebijakan bucket ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

JSON

Example : kebijakan bucket S3 pada Outposts dengan pengguna utama wildcard (*****) dan kunci syarat untuk membatasi akses ke rentang alamat IP tertentu

Kebijakan bucket berikut menggunakan pengguna utama wildcard (*) dengan syarat `aws:SourceIp` untuk membatasi akses ke rentang alamat IP tertentu. Untuk menggunakan kebijakan bucket ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "ExampleBucketPolicy2",
    "Statement": [
        {
            "Sid": "statement1",
            "Effect": "Allow",
            "Principal": { "AWS" : "*" },
```

```
            "Action": [
                "s3-outposts:GetObject",
                "s3-outposts:PutObject",
                "s3-outposts:DeleteObject",
                "s3-outposts:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3-outposts:aws-
region:123456789012:outpost/op-01ac5d28a6a232904/bucket/",
                "arn:aws:s3-outposts:aws-
region:123456789012:outpost/op-01ac5d28a6a232904/bucket//*"
            ],
            "Condition" : {
                "IpAddress" : {
                    "aws:SourceIp": "192.0.2.0/24"
                },
                "NotIpAddress" : {
                    "aws:SourceIp": "198.51.100.0/24"
                }
            }
        }
    ]
}
```

Izin untuk titik akhir S3 di Outposts

S3 di Outposts memerlukan izinnya sendiri di IAM untuk mengelola tindakan titik akhir S3 di Outposts.

> **Note**
>
> - Untuk titik akhir yang menggunakan jenis akses kumpulan alamat IP (kumpulan CoIP) milik pelanggan, Anda juga harus memiliki izin untuk bekerja dengan alamat IP dari kumpulan CoIP Anda, seperti yang dijelaskan dalam tabel berikut.
> - Untuk akun bersama yang mengakses S3 di Outposts dengan AWS Resource Access Manager menggunakan, pengguna di akun bersama ini tidak dapat membuat titik akhir mereka sendiri di subnet bersama. Apabila pengguna di akun bersama ingin mengelola titik

akhir mereka sendiri, akun bersama harus membuat subnetnya sendiri di Outpost. Untuk informasi selengkapnya, lihat the section called “Berbagi S3 di Outposts”.

Tabel berikut menunjukkan S3 pada izin IAM terkait titik akhir Outposts.

Tindakan	Izin IAM

`CreateEndpoint`	`s3-outposts:CreateEndpoint`
	`ec2:CreateNetworkInterface`
	`ec2:DescribeNetworkInterfaces`
	`ec2:DescribeVpcs`
	`ec2:DescribeSecurityGroups`
	`ec2:DescribeSubnets`
	`ec2:CreateTags`
	`iam:CreateServiceLinkedRole`
	Untuk titik akhir yang menggunakan jenis akses kumpulan alamat IP (kumpulan CoIP) milik pelanggan on-premise, izin tambahan berikut diperlukan:
	`s3-outposts:CreateEndpoint`
	`ec2:DescribeCoipPools`
	`ec2:GetCoipPoolUsage`
	`ec2:AllocateAddress`
	`ec2:AssociateAddress`
	`ec2:DescribeAddresses`

Tindakan	Izin IAM
	`ec2:DescribeLocalGatewayRouteTableVpcAssociations`

`DeleteEndpoint`	`s3-outposts:DeleteEndpoint`
	
	`ec2:DeleteNetworkInterface`
	
	`ec2:DescribeNetworkInterfaces`
	
	Untuk titik akhir yang menggunakan jenis akses kumpulan alamat IP (kumpulan CoIP) milik pelanggan on-premise, izin tambahan berikut diperlukan:
	
	`s3-outposts:DeleteEndpoint`
	
	`ec2:DisassociateAddress`
	
	`ec2:DescribeAddresses`
	
	`ec2:ReleaseAddress`

`ListEndpoints`	`s3-outposts:ListEndpoints`

> **Note**
>
> Anda dapat menggunakan tag sumber daya dalam kebijakan IAM untuk mengelola izin.

Peran yang ditautkan dengan layanan untuk S3 di Outposts

S3 di Outposts menggunakan peran yang ditautkan dengan layanan IAM untuk membuat beberapa sumber daya jaringan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan Peran Terkait Layanan untuk Amazon S3 di Outposts.

Enkripsi data dalam S3 di Outposts

Secara default, semua data yang disimpan dalam Amazon S3 di Outposts dienkripsi menggunakan enkripsi di sisi server dengan kunci enkripsi yang dikelola Amazon S3 (SSE-S3). Untuk informasi selengkapnya, lihat Menggunakan enkripsi sisi server dengan kunci terkelola Amazon S3 (SSE-S3) di Panduan Pengguna Amazon S3.

Anda juga dapat menggunakan enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C). Untuk menggunakan SSE-C, tentukan kunci enkripsi sebagai bagian dari permintaan API objek Anda. Enkripsi di sisi server hanya mengenkripsi data objek, bukan metadata objek. Untuk informasi selengkapnya, lihat Menggunakan enkripsi sisi server dengan kunci yang disediakan pelanggan di Panduan Pengguna Amazon S3.

> **Note**
>
> S3 di Outposts tidak mendukung enkripsi sisi server AWS Key Management Service dengan kunci AWS KMS() (SSE-KMS).

AWS PrivateLink untuk S3 di Outposts

S3 on Outposts AWS PrivateLink mendukung, yang menyediakan akses manajemen langsung ke S3 Anda pada penyimpanan Outposts melalui titik akhir pribadi dalam jaringan pribadi virtual Anda. Ini memungkinkan Anda untuk menyederhanakan arsitektur jaringan internal Anda dan melakukan operasi manajemen pada penyimpanan objek Outposts Anda dengan menggunakan alamat IP pribadi di Virtual Private Cloud (VPC) Anda. Menggunakan AWS PrivateLink menghilangkan kebutuhan untuk menggunakan alamat IP publik atau server proxy.

Dengan AWS PrivateLink Amazon S3 di Outposts, Anda dapat menyediakan titik akhir VPC antarmuka di cloud pribadi virtual (VPC) Anda untuk mengakses S3 pada manajemen bucket Outposts dan manajemen endpoint. APIs Titik akhir VPC antarmuka dapat diakses langsung dari aplikasi yang digunakan di VPC Anda atau di tempat melalui jaringan privat virtual (VPN) Anda atau. AWS Direct Connect Anda dapat mengakses bucket dan manajemen endpoint APIs melalui AWS PrivateLink. AWS PrivateLink tidak mendukung operasi API transfer data, seperti GET, PUT, dan sejenisnya APIs. Operasi ini sudah ditransfer secara pribadi melalui titik akhir S3 pada Outposts dan konfigurasi titik akses. Untuk informasi selengkapnya, lihat Jaringan untuk S3 di Outposts.

Endpoint antarmuka diwakili oleh satu atau lebih antarmuka jaringan elastis (ENIs) yang diberi alamat IP pribadi dari subnet di VPC Anda. Permintaan yang dibuat untuk menghubungkan titik akhir untuk S3 di Outposts secara otomatis dirutekan ke S3 pada bucket Outposts dan manajemen endpoint di jaringan. APIs AWS Anda juga dapat mengakses titik akhir antarmuka di VPC Anda dari aplikasi lokal AWS Direct Connect melalui AWS Virtual Private Network atau ().AWS VPN Untuk informasi selengkapnya tentang cara menghubungkan VPC dengan jaringan on-premise Anda, lihat AWS Direct Connect Panduan Pengguna dan AWS Site-to-Site VPN Panduan Pengguna.

Permintaan rute titik akhir antarmuka untuk S3 pada bucket Outposts dan manajemen APIs endpoint melalui AWS jaringan dan melalui AWS PrivateLink, seperti yang diilustrasikan dalam diagram berikut.

Untuk informasi umum tentang titik akhir antarmuka, lihat Antarmuka titik akhir VPC (AWS PrivateLink) dalam Panduan AWS PrivateLink .

Topik

- Pembatasan dan batasan
- Mengakses S3 di Outposts
- Memperbarui konfigurasi DNS on-premise
- Membuat titik akhir VPC untuk S3 di Outposts
- Membuat kebijakan bucket dan kebijakan titik akhir VPC untuk S3 di Outposts

Pembatasan dan batasan

Saat Anda mengakses S3 di bucket Outposts dan manajemen endpoint APIs melalui AWS PrivateLink, pembatasan VPC berlaku. Untuk informasi selengkapnya, lihat Properti titik akhir antarmuka dan batas dan AWS PrivateLink kuota di AWS PrivateLink Panduan.

Selain itu, AWS PrivateLink tidak mendukung yang berikut:

- Titik Akhir Standar Proses Informasi Federal (FIPS)
- S3 pada APIs transfer data Outposts, misalnya, GET, PUT, dan operasi API objek serupa.
- DNS privat

Mengakses S3 di Outposts

Untuk mengakses S3 pada bucket Outposts dan APIs manajemen endpoint AWS PrivateLink menggunakan, Anda harus memperbarui aplikasi Anda untuk menggunakan nama DNS khusus titik akhir. Saat Anda membuat titik akhir antarmuka, AWS PrivateLink buat dua jenis S3 khusus titik akhir pada nama Outposts: Regional dan zonal.

- Nama DNS regional — termasuk ID titik akhir VPC unik, pengenal layanan, `vpce.amazonaws.com` dan, Wilayah AWS misalnya,. *`vpce-1a2b3c4d-5e6f`*`.s3-outposts.us-east-1.vpce.amazonaws.com`
- Nama DNS zona — termasuk ID titik akhir VPC unik, Availability Zone, pengenal layanan, dan, misalnya Wilayah AWS,. `vpce.amazonaws.com` *`vpce-1a2b3c4d-5e6f`*`-us-east-1a.s3-outposts.us-east-1.vpce.amazonaws.com` Anda dapat menggunakan opsi ini jika arsitektur Anda mengisolasi Zona Ketersediaan. Misalnya, Anda bisa menggunakan nama DNS zonal untuk penahanan kesalahan atau untuk mengurangi biaya transfer data Regional.

> **Important**
>
> S3 pada titik akhir antarmuka Outposts diselesaikan dari domain DNS publik. S3 di Outposts tidak mendukung DNS pribadi. Gunakan `--endpoint-url` parameter untuk semua manajemen APIs bucket dan endpoint.

AWS CLI contoh

Gunakan `--endpoint-url` parameter `--region` dan untuk mengakses manajemen bucket dan manajemen endpoint APIs melalui S3 pada titik akhir antarmuka Outposts.

Example : Gunakan URL titik akhir untuk daftar bucket dengan API kontrol S3

Dalam contoh berikut, ganti Wilayah *`us-east-1`*, URL titik akhir VPC*`vpce-1a2b3c4d-5e6f.s3.us-east-1.vpce.amazonaws.com`*, dan ID akun *`111122223333`* dengan informasi yang sesuai.

```
aws s3control list-regional-buckets --region us-east-1 --endpoint-url
 https://vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com --account-
id 111122223333
```

AWS Contoh SDK

Perbarui SDKs ke versi terbaru, dan konfigurasikan klien Anda untuk menggunakan URL titik akhir untuk mengakses API kontrol S3 untuk S3 pada titik akhir antarmuka Outposts.

SDK for Python (Boto3)

Example : Gunakan URL titik akhir untuk mengakses API kontrol S3

Dalam contoh berikut, ganti URL titik akhir Wilayah *us-east-1* dan VPC *vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com* dengan informasi yang sesuai.

```
control_client = session.client(
service_name='s3control',
region_name='us-east-1',
endpoint_url='https://vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com'
)
```

Untuk informasi selengkapnya, lihat AWS PrivateLink Amazon S3 di panduan pengembang Boto3.

SDK for Java 2.x

Example : Gunakan URL titik akhir untuk mengakses API kontrol S3

Dalam contoh berikut, ganti URL titik akhir VPC *vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com* dan Wilayah *Region.US_EAST_1* dengan informasi yang sesuai.

```
// control client
Region region = Region.US_EAST_1;
s3ControlClient = S3ControlClient.builder().region(region)

 .endpointOverride(URI.create("https://vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com"))
                                .build()
```

Untuk informasi selengkapnya, lihat `S3ControlClient` dalam Referensi API AWS SDK untuk Java .

Memperbarui konfigurasi DNS on-premise

Saat menggunakan nama DNS khusus titik akhir untuk mengakses titik akhir antarmuka untuk S3 pada manajemen bucket Outposts dan manajemen titik akhir APIs, Anda tidak perlu memperbarui penyelesai DNS lokal. Anda dapat menyelesaikan nama DNS titik akhir khusus dengan alamat IP privat titik akhir antarmuka dari domain S3 publik di Outposts.

Membuat titik akhir VPC untuk S3 di Outposts

Untuk membuat titik akhir antarmuka VPC untuk S3 di Outposts, lihat Membuat titik akhir VPC di Panduan.AWS PrivateLink

Membuat kebijakan bucket dan kebijakan titik akhir VPC untuk S3 di Outposts

Anda dapat melampirkan kebijakan titik akhir ke titik akhir VPC yang mengontrol akses ke S3 di Outposts. Anda juga dapat menggunakan kebijakan `aws:sourceVpce` bucket S3 di Outposts untuk membatasi akses ke bucket tertentu dari titik akhir VPC tertentu. Dengan kebijakan titik akhir VPC, Anda dapat mengontrol akses ke S3 pada manajemen bucket Outposts dan manajemen endpoint. APIs APIs Dengan kebijakan bucket, Anda dapat mengontrol akses ke manajemen bucket S3 on Outposts. APIs Namun, Anda tidak dapat mengelola akses ke tindakan objek untuk S3 di `aws:sourceVpce` Outposts menggunakan.

Kebijakan akses untuk S3 di Outposts menentukan informasi berikut:

- Prinsip AWS Identity and Access Management (IAM) yang tindakannya diizinkan atau ditolak.
- Tindakan kontrol S3 yang diizinkan atau ditolak.
- Sumber daya S3 di Outposts di mana tindakan diizinkan atau ditolak.

Contoh berikut menunjukkan kebijakan yang membatasi akses ke bucket atau titik akhir. Untuk informasi selengkapnya tentang konektivitas VPC, lihat opsi konektivitas di AWS whitepaper Opsi Network-to-VPC Konektivitas Amazon Virtual Private Cloud.

> **Important**
>
> - Saat menerapkan kebijakan contoh untuk titik akhir VPC yang dijelaskan di bagian ini, Anda dapat memblokir akses Anda ke bucket tanpa bermaksud melakukannya. Izin bucket

yang membatasi akses bucket ke koneksi yang berasal dari titik akhir VPC Anda dapat memblokir semua koneksi ke bucket tersebut. Untuk informasi tentang cara mengatasi masalah ini, lihat Kebijakan bucket saya memiliki ID titik akhir VPC atau VPC yang salah. Bagaimana saya dapat memperbaiki kebijakan tersebut sehingga saya dapat mengakses bucket? dalam Pusat Pengetahuan Dukungan .

- Sebelum menggunakan kebijakan bucket contoh berikut ini, ganti ID titik akhir VPC dengan nilai yang sesuai untuk kasus penggunaan Anda. Jika tidak, Anda tidak akan dapat mengakses bucket Anda.
- Jika kebijakan Anda hanya mengizinkan akses ke bucket S3 di Outposts dari titik akhir VPC tertentu, kebijakan tersebut nonaktifkan akses konsol untuk ember tersebut karena permintaan konsol tidak berasal dari titik akhir VPC tertentu.

Topik

Contoh: Membatasi Akses ke bucket khusus dari titik akhir VPC

Anda dapat membuat kebijakan titik akhir yang membatasi akses ke bucket S3 tertentu di Outposts saja. Kebijakan berikut membatasi akses untuk GetBucketPolicy tindakan hanya ke. *example-outpost-bucket* Untuk menggunakan kebijakan ini, ganti nilai contoh dengan kebijakan Anda sendiri.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "Policy1415115909151",
    "Statement": [
        {
            "Sid": "Access-to-specific-bucket-only",
            "Principal": {
                "AWS": "111122223333"
            },
            "Action": "s3-outposts:GetBucketPolicy",
            "Effect": "Allow",
```

```
            "Resource": "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket"
        }
    ]
}
```

Contoh: Menolak Akses dari titik akhir VPC tertentu di kebijakan bucket S3 di Outposts

Kebijakan bucket S3 on Outposts berikut menolak akses GetBucketPolicy ke bucket melalui endpoint *example-outpost-bucket* VPC. *vpce-1a2b3c4d*

`aws:sourceVpce`Syarat menentukan titik akhir dan tidak memerlukan Amazon Resource Name (ARN) untuk sumber daya titik akhir VPC, tetapi hanya ID titik akhir. Untuk menggunakan kebijakan ini, ganti nilai contoh dengan kebijakan Anda sendiri.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "Policy1415115909152",
    "Statement": [
        {
            "Sid": "Deny-access-to-specific-VPCE",
            "Principal": {
                "AWS": "111122223333"
            },
            "Action": "s3-outposts:GetBucketPolicy",
            "Effect": "Deny",
            "Resource": "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpce": "vpce-1a2b3c4d"
                }
            }
        }
    ]
}
```

AWS Kunci kebijakan khusus otentikasi Versi Tanda Tangan 4 (SiGv4)

Tabel berikut menunjukkan kunci kondisi yang terkait dengan otentikasi AWS Signature Version 4 (SigV4) yang dapat Anda gunakan dengan Amazon S3 di Outposts. Dalam kebijakan bucket, Anda dapat menambahkan kondisi ini untuk menerapkan perilaku tertentu saat permintaan diautentikasi menggunakan Signature Versi 4. Untuk kebijakan-kebijakan contoh, lihat Contoh kebijakan bucket yang menggunakan kunci kondisi terkait Signature Version 4. Untuk informasi selengkapnya tentang mengautentikasi permintaan menggunakan Signature Versi 4, lihat Mengautentikasi permintaan (Versi AWS Tanda Tangan 4) di Referensi API Amazon Simple Storage Service

Kunci yang berlaku	Deskripsi

`s3-outposts:authType`	S3 di Outposts mendukung berbagai metode autentikasi. Untuk membatasi permintaan masuk untuk menggunakan metode autentikasi tertentu, Anda dapat menggunakan kunci syarat opsional ini. Misalnya, Anda dapat menggunakan kunci syarat ini untuk mengizinkan hanya header `Authorization` HTTP yang digunakan dalam autentikasi permintaan.
	
	Nilai valid:
	
	`REST-HEADER`
	
	`REST-QUERY-STRING`

`s3-outposts:signatureAge`	Lamanya waktu, dalam milidetik, tanda tangan valid dalam permintaan autentikasi. Kondisi ini hanya berfungsi untuk presigned URLs. Di Signature Versi 4, kunci penandatangan berlaku hingga tujuh hari. Karena itu, tanda tangan juga berlaku hingga tujuh hari. Untuk informasi selengkapnya, lihat Pendahuluan ke permintaan penandatanganan dalam Referensi API Amazon Simple Storage Service. Anda dapat menggunakan syarat ini untuk lebih lanjut membatasi umur tanda tangan. Nilai contoh: `600000`

Kunci yang berlaku	Deskripsi
`s3-outposts:x-amz-content-sha256`	Anda dapat menggunakan kunci syarat ini untuk melarang konten yang tidak ditandatangani di bucket Anda. Ketika Anda menggunakan Signature Version 4 untuk permintaan yang menggunakan `Authorization` header Authorization, Anda menambahka `x-amz-content-sha256` header x-amz-content-sha256 dalam perhitungan tanda tangan, lalu kemudian menetapkan nilainya ke muatan hash. Anda dapat menggunakan kunci kondisi ini dalam kebijakan bucket untuk menolak setiap unggahan yang tidak ditandatangani payload. Sebagai contoh: • Menolak unggahan yang menggunakan header `Authorization` untuk mengautentikasi permintaan tetapi tidak menandatangani muatan. Untuk informasi selengkapnya, lihat Memindahkan muatan dalam satu bagian tunggal di Referensi API Amazon Simple Storage Service. • Tolak unggahan yang menggunakan URLs presigned. Presigned URLs selalu memiliki`UNSIGNED_PAYLOAD` . Untuk informasi selengkap nya, lihat Mengautentikasi permintaan dan Metode autentikasi dalam Referensi API Amazon Simple Storage Service. Nilai valid: `UNSIGNED-PAYLOAD`

Contoh kebijakan bucket yang menggunakan kunci kondisi terkait Signature Version 4

Untuk menggunakan contoh berikut, ganti *`user input placeholders`*dengan informasi Anda sendiri.

Example : **`s3-outposts:signatureAge`**

Kebijakan bucket berikut menolak permintaan URL yang telah ditetapkan sebelumnya pada S3 pada Outposts pada objek `example-outpost-bucket` jika tanda tangan berusia lebih dari 10 menit.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Deny a presigned URL request if the signature is more than 10
 minutes old",
            "Effect": "Deny",
            "Principal": {"AWS":"444455556666"},
            "Action": "s3-outposts:*",
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
            "Condition": {
                "NumericGreaterThan": {"s3-outposts:signatureAge": 600000},
                "StringEquals": {"s3-outposts:authType": "REST-QUERY-STRING"}
            }
        }
    ]
}
```

Example : **`s3-outposts:authType`**

Kebijakan bucket berikut hanya mengizinkan permintaan yang menggunakan `Authorization` header untuk otentikasi permintaan. Permintaan URL yang telah ditetapkan sebelumnya akan ditolak karena parameter kueri URLs penggunaan yang telah ditetapkan sebelumnya untuk memberikan informasi permintaan dan otentikasi. Untuk informasi selengkapnya, lihat Metode autentikasi di Referensi API Amazon Simple Storage Service.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
         {
               "Sid": "Allow only requests that use the Authorization header for
 request authentication. Deny presigned URL requests.",
               "Effect": "Deny",
               "Principal": {"AWS":"111122223333"},
```

```
                "Action": "s3-outposts:*",
                "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
                "Condition": {
                      "StringNotEquals": {
                            "s3-outposts:authType": "REST-HEADER"
                      }
                }
          }
   ]
}
```

Example : **s3-outposts:x-amz-content-sha256**

Kebijakan bucket berikut menolak setiap unggahan dengan muatan yang tidak ditandatangani, seperti unggahan yang menggunakan presigned. URLs Untuk informasi selengkapnya, lihat Mengautentikasi permintaan dan Metode autentikasi dalam Referensi API Amazon Simple Storage Service.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
          {
                "Sid": "Deny uploads with unsigned payloads.",
                "Effect": "Deny",
                "Principal": {"AWS":"111122223333"},
                "Action": "s3-outposts:*",
                "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
                "Condition": {
                      "StringEquals": {
                            "s3-outposts:x-amz-content-sha256": "UNSIGNED-PAYLOAD"
                      }
                }
          }
   ]
}
```

AWS kebijakan terkelola untuk Amazon S3 di Outposts

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa kebijakan AWS terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk digunakan semua pelanggan. AWS Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Jika AWS memperbarui izin yang ditentukan dalam kebijakan AWS terkelola, pembaruan akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat Kebijakan terkelola AWS dalam Panduan Pengguna IAM.

AWS kebijakan terkelola: AWSS3 OnOutpostsServiceRolePolicy

Membantu mengelola sumber daya jaringan untuk Anda sebagai bagian dari peran terkait layanan `AWSServiceRoleForS3OnOutposts`.

Untuk melihat izin kebijakan ini, lihat AWSS3OnOutpostsServiceRolePolicy.

Pembaruan AWS S3 tentang Outposts ke kebijakan terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk S3 di Outposts sejak layanan ini mulai melacak perubahan ini.

Perubahan	Deskripsi	Tanggal
S3 di Outposts menambahkan `AWSS3OnOutpostsServiceRolePolicy`	S3 di Outposts menambahkan `AWSS3OnOutpostsServiceRolePolicy` sebagai bagian dari peran terkait layanan `AWSServiceRoleForS3OnOutposts` , yang membantu	3 Oktober 2023

Perubahan	Deskripsi	Tanggal
	mengelola sumber daya jaringan untuk Anda.	
S3 di Outposts mulai melacak perubahan	S3 di Outposts mulai melacak perubahan untuk AWS kebijakan yang dikelola.	3 Oktober 2023

Menggunakan Peran Terkait Layanan untuk Amazon S3 di Outposts

Amazon S3 di Outposts menggunakan peran terkait layanan AWS Identity and Access Management (IAM). Peran terkait layanan adalah tipe peran IAM unik yang terhubung langsung ke S3 di Outposts. Peran terkait layanan telah ditentukan sebelumnya oleh S3 di Outposts dan menyertakan semua izin yang diperlukan layanan untuk memanggil layanan lain atas nama Anda. AWS

Peran terkait layanan mempermudah pengaturan S3 di Outposts karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. S3 di Outposts menentukan izin peran terkait layanan, kecuali jika ditentukan lain, hanya S3 di Outposts yang dapat mengasumsikan peran tersebut. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus sumber daya terkait terlebih dahulu. Ini melindungi sumber daya S3 di Outposts Anda karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat Layanan AWS yang Bekerja bersama IAM dan mencari layanan yang memiliki Ya dalam Peran Terkait Layanan. Pilih Ya bersama tautan untuk melihat dokumentasi peran tertaut layanan untuk layanan tersebut.

Izin peran terkait layanan untuk S3 di Outposts

S3 di Outposts menggunakan peran terkait layanan AWSServiceRoleForbernama OnOutposts S3 untuk membantu mengelola sumber daya jaringan untuk Anda.

`AWSServiceRoleForS3OnOutposts` peran terkait layanan memercayakan layanan berikut untuk menjalankan peran tersebut:

- `s3-outposts.amazonaws.com`

Kebijakan izin peran yang diberi nama `AWSS3OnOutpostsServiceRolePolicy` memungkinkan S3 di Outposts untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSubnets",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeVpcs",
                "ec2:DescribeCoipPools",
                "ec2:GetCoipPoolUsage",
                "ec2:DescribeAddresses",
                "ec2:DescribeLocalGatewayRouteTableVpcAssociations"
            ],
            "Resource": "*",
            "Sid": "DescribeVpcResources"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ],
            "Sid": "CreateNetworkInterface"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "CreateTagsForCreateNetworkInterface"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:ipv4pool-ec2/*"
            ],
            "Sid": "AllocateIpAddress"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:elastic-ip/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "CreateTagsForAllocateIpAddress"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DisassociateAddress",
                "ec2:ReleaseAddress",
                "ec2:AssociateAddress"
            ],
```

```
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "ReleaseVpcResources"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": [
                        "CreateNetworkInterface",
                        "AllocateAddress"
                    ],
                    "aws:RequestTag/CreatedBy": [
                        "S3 On Outposts"
                    ]
                }
            },
            "Sid": "CreateTags"
        }
    ]
}
```

Anda harus mengonfigurasikan izin untuk mengizinkan entitas IAM (seperti peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat Izin peran tertaut layanan dalam Panduan Pengguna IAM.

Membuat Peran Terkait Layanan untuk S3 di Outposts

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda membuat titik akhir S3 di Outposts di, AWS Management Console AWS CLI the, atau AWS API, S3 di Outposts membuat peran terkait layanan untuk Anda.

Jika Anda menghapus peran tertaut layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda membuat titik akhir S3 di Outposts menciptakan peran terkait layanan untuk Anda.

Anda juga dapat menggunakan konsol IAM untuk membuat peran terkait layanan dengan kasus penggunaan S3 di Outposts. Di AWS CLI atau AWS API, buat peran terkait layanan dengan nama `s3-outposts.amazonaws.com` layanan. Untuk informasi selengkapnya, lihat Membuat peran tertaut layanan dalam Panduan Pengguna IAM. Jika Anda menghapus peran tertaut layanan ini, Anda dapat mengulang proses yang sama untuk membuat peran tersebut lagi.

Mengedit Peran Terkait Layanan untuk S3 di Outposts

S3 di Outposts tidak mengizinkan Anda untuk mengedit peran terkait layanan`AWSServiceRoleForS3OnOutposts`. Ini termasuk nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran tertaut layanan dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk S3 di Outposts

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, kami merekomendasikan Anda menghapus peran tersebut. Dengan begitu, Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Tetapi, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

> **Note**
>
> Jika layanan S3 di Outposts menggunakan peran saat Anda mencoba untuk menghapus sumber daya, maka penghapusan tersebut kemungkinan gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Untuk menghapus S3 pada sumber daya Outposts yang digunakan oleh AWSService RoleFor peran S3 OnOutposts

1. Hapus titik akhir S3 di Outposts di Akun AWS seluruh Anda. Wilayah AWS
2. Hapus peran terkait layanan menggunakan IAM.

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForS3OnOutposts` terkait layanan. Untuk informasi selengkapnya, lihat Menghapus peran tertaut layanan dalam Panduan Pengguna IAM.

Wilayah yang Di Support untuk S3 di Peran Terkait Layanan Outposts

S3 on Outposts mendukung penggunaan peran terkait layanan di semua tempat layanan Wilayah AWS tersedia. Untuk informasi lebih lanjut, lihat S3 di Wilayah Outposts dan titik akhir.

Mengelola penyimpanan S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Untuk informasi selengkapnya tentang pengelolaan dan berbagi Amazon S3 di Outposts, lihat topik berikut.

Topik

- Mengelola Penentuan Versi untuk bucket S3 di Outposts
- Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda
- Replikasi objek untuk S3 di Outposts.
- Berbagi S3 di Outposts dengan menggunakan AWS RAM
- Lainnya Layanan AWS yang menggunakan S3 di Outposts

Mengelola Penentuan Versi untuk bucket S3 di Outposts

Saat diaktifkan, S3 Penentuan Versi menyimpan beberapa salinan objek yang berbeda dalam bucket yang sama. Anda dapat menggunakan penentuan versi S3 untuk menyimpan, mengambil, dan memulihkan setiap versi dari setiap objek yang disimpan dalam bucket Outposts Anda. Penentuan Versi S3 membantu Anda memulihkan dari tindakan pengguna yang tidak diinginkan dan kegagalan aplikasi.

Bucket Amazon S3 di Outposts memiliki tiga status Penentuan Versi:

- Tidak Berversi-Jika Anda belum pernah mengaktifkan atau menangguhkan Pembuatan Versi S3 di bucket Anda, itu tidak berversi dan tidak mengembalikan status Pembuatan Versi S3. Untuk

informasi selengkapnya tentang Penentuan Versi S3, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.

- Diaktifkan-Mengaktifkan Penentuan Versi S3 untuk objek di bucket. Semua objek yang ditambahkan ke bucket menerima ID versi unik. Objek yang sudah ada di bucket pada saat Anda mengaktifkan Penentuan Versi memiliki ID versi `null`. Jika Anda memodifikasi objek ini (atau lainnya) dengan operasi lain PutObject, seperti, objek baru mendapatkan ID versi unik.
- Ditangguhkan-Penentuan Versi S3 untuk objek di bucket. Semua objek yang ditambahkan ke bucket setelah penentuan versi ditangguhkan menerima ID versi `null`. Untuk informasi selengkapnya, lihat Menambahkan objek ke bucket yang ditangguhkan versi di Panduan Pengguna Amazon S3.

Setelah Anda mengaktifkan Penentuan Versi untuk bucket S3 di Outposts, itu tidak akan pernah dapat kembali ke status tidak berversi. Namun, Anda dapat menangguhkan Penentuan Versi. Untuk informasi selengkapnya tentang Penentuan Versi S3, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.

Untuk setiap objek dalam bucket, Anda memiliki versi saat ini dan nol atau lebih versi non-terkini. Untuk mengurangi biaya penyimpanan, Anda dapat mengonfigurasi aturan Siklus Hidup bucket S3 agar versi noncurrent kedaluwarsa setelah periode waktu yang ditentukan. Untuk informasi selengkapnya, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.

Contoh berikut menunjukkan cara mengaktifkan atau menangguhkan pembuatan versi untuk bucket S3 di Outposts yang ada dengan menggunakan AWS Management Console dan (). AWS Command Line Interface AWS CLI Untuk membuat bucket dengan Penentuan Versi S3, lihat. Membuat bucket S3 di Outposts

> **Note**
>
> Akun AWS Yang menciptakan ember memilikinya dan merupakan satu-satunya yang dapat melakukan tindakan untuk itu. Bucket memiliki properti konfigurasi, seperti Outpost, tag, enkripsi default, dan pengaturan titik akses. Pengaturan titik akses termasuk cloud privat virtual (VPC), kebijakan titik akses untuk mengakses objek di bucket, dan metadata lainnya. Untuk informasi selengkapnya, lihat Spesifikasi S3 di Outposts.

Menggunakan konsol S3

Untuk mengedit setelan Pembuatan Versi S3 untuk bucket

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda aktifkan Penentuan Versi untuk S3.
4. Pilih tab Properti.
5. Di bawah Penentuan Versi Bucket, pilih Edit.
6. Edit pengaturan S3 Penentuan Versi untuk bucket dengan memilih salah satu opsi berikut:
   - Untuk menangguhkan Pembuatan Versi S3 dan menghentikan pembuatan versi objek baru, pilih Tangguhkan.
   - Untuk mengaktifkan S3 Penentuan Versi dan menyimpan beberapa salinan berbeda dari setiap objek, pilih Aktifkan.
7. Pilih Simpan perubahan.

Menggunakan AWS CLI

Untuk mengaktifkan atau menangguhkan Pembuatan Versi S3 untuk bucket Anda dengan menggunakan AWS CLI, gunakan `put-bucket-versioning` perintah, seperti yang ditunjukkan pada contoh berikut. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

Untuk informasi selengkapnya, lihat put-bucket-versioning dalam Referensi AWS CLI .

Example : Untuk mengaktifkan S3 Penentuan Versi

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket --versioning-configuration Status=Enabled
```

Example : Untuk menangguhkan Pembuatan Versi S3

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket --versioning-configuration Status=Suspended
```

Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda

Anda dapat menggunakan Siklus Hidup S3 untuk mengoptimalkan kapasitas penyimpanan Amazon S3 di Outposts. Anda dapat membuat aturan siklus hidup untuk menjadikan objek kedaluwarsa seiring bertambahnya umur objek atau digantikan oleh versi yang lebih baru. Anda dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk informasi selengkapnya tentang Siklus Hidup S3, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.

> **Note**
>
> Akun AWS Yang membuat bucket memilikinya dan merupakan satu-satunya yang dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk membuat dan mengelola konfigurasi siklus hidup bucket S3 di Outposts Anda, lihat topik berikut.

Topik

- Membuat dan mengelola aturan siklus hidup dengan menggunakan AWS Management Console
- Membuat dan mengelola konfigurasi siklus hidup dengan menggunakan AWS CLI dan SDK for Java

Membuat dan mengelola aturan siklus hidup dengan menggunakan AWS Management Console

Anda dapat menggunakan Siklus Hidup S3 untuk mengoptimalkan kapasitas penyimpanan Amazon S3 di Outposts. Anda dapat membuat aturan siklus hidup untuk menjadikan objek kedaluwarsa seiring bertambahnya umur objek atau digantikan oleh versi yang lebih baru. Anda dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk informasi selengkapnya tentang Siklus Hidup S3, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.

> **Note**
>
> Akun AWS Yang membuat bucket memilikinya dan merupakan satu-satunya yang dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk membuat dan mengelola aturan siklus hidup untuk S3 di Outposts menggunakan AWS Management Console, lihat topik berikut.

Topik

- Membuat aturan siklus hidup
- Mengaktifkan aturan siklus hidup
- Mengedit aturan siklus hidup
- Menghapus aturan siklus hidup

Membuat aturan siklus hidup

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda buat aturan siklus hidupnya.
4. Pilih tab Manajemen, kemudian pilih Buat aturan siklus hidup.
5. Masukkan nilai untuk Nama aturan siklus hidup.
6. Dalam Cakupan aturan, pilih salah satu opsi berikut:
   - Untuk membatasi cakupan pada filter tertentu, pilih Batasi cakupan aturan ini menggunakan satu atau beberapa filter. Kemudian, tambahkan filter prefiks, tag, atau ukuran objek.
   - Untuk menerapkan aturan pada semua objek dalam bucket, pilih Terapkan pada semua objek dalam bucket.
7. Dalam Tindakan aturan siklus hidup, pilih salah satu opsi berikut:
   - Jadikan versi objek saat ini kedaluwarsa–Untuk bucket berkemampuan Penentuan Versi, S3 di Outposts menambahkan penanda hapus dan mempertahankan objek sebagai versi tidak terkini. Untuk bucket yang tidak menggunakan Penentuan Versi S3, S3 di Outposts menghapus objek secara permanen.

- Hapus secara permanen versi objek tidak terkini –S3 di Outposts secara permanen menghapus versi objek tidak terkini.
- Hapus penanda hapus objek kedaluwarsa atau unggahan multibagian yang tidak lengkap–S3 di Outposts secara permanen menghapus penanda hapus objek kedaluwarsa atau unggahan multibagian yang tidak lengkap.

  Apabila Anda membatasi cakupan aturan Siklus Hidup menggunakan tag objek, Anda tidak dapat memilih Hapus penanda hapus objek kedaluwarsa. Anda juga tidak dapat memilih Hapus penanda hapus objek kedaluwarsa jika Anda memilih Jadikan versi objek saat ini kedaluwarsa.

> **Note**
>
> Filter berbasis ukuran tidak dapat digunakan dengan penanda hapus dan unggahan multibagian yang tidak lengkap.

8. Apabila Anda memilih Jadikan versi objek saat ini kedaluwarsa atau Hapus versi objek yang tidak aktif secara permanen, konfigurasikan pemicu aturan berdasarkan tanggal tertentu atau umur objek.
9. Apabila Anda memilih Hapus penanda hapus objek kedaluwarsa, untuk mengonfirmasi bahwa Anda ingin menghapus penanda hapus objek kedaluwarsa, pilih Hapus penanda hapus objek kedaluwarsa.
10. Dalam Ringkasan Lini Masa, tinjau aturan Siklus Hidup Anda, dan pilih Buat aturan.

Mengaktifkan aturan siklus hidup

Untuk mengaktifkan atau menonaktifkan aturan siklus hidup bucket

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda aktifkan atau nonaktifkan aturan siklus hidupnya.
4. Pilih tab Manajemen, kemudian dalam Aturan Siklus Hidup, pilih aturan yang ingin Anda aktifkan atau nonaktifkan.
5. Untuk Tindakan, pilih Aktifkan atau nonaktifkan aturan.

Mengedit aturan siklus hidup

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda edit aturan siklus hidupnya.
4. Pilih tab Manajemen, kemudian pilih Aturan Siklus Hidup yang ingin Anda edit.
5. (Opsional) Perbarui nilai untuk nama aturan Siklus hidup.
6. Dalam Cakupan aturan, edit cakupan sesuai kebutuhan:

   - Untuk membatasi cakupan pada filter tertentu, pilih Batasi cakupan aturan ini menggunakan satu atau beberapa filter. Kemudian, tambahkan filter prefiks, tag, atau ukuran objek.
   - Untuk menerapkan aturan pada semua objek dalam bucket, pilih Terapkan pada semua objek dalam bucket.
7. Dalam Tindakan aturan siklus hidup, pilih salah satu opsi berikut:

   - Jadikan versi objek saat ini kedaluwarsa–Untuk bucket berkemampuan Penentuan Versi, S3 di Outposts menambahkan penanda hapus dan mempertahankan objek sebagai versi tidak terkini. Untuk bucket yang tidak menggunakan Penentuan Versi S3, S3 di Outposts menghapus objek secara permanen.
   - Hapus secara permanen versi objek tidak terkini –S3 di Outposts secara permanen menghapus versi objek tidak terkini.
   - Hapus penanda hapus objek kedaluwarsa atau unggahan multibagian yang tidak lengkap–S3 di Outposts secara permanen menghapus penanda hapus objek kedaluwarsa atau unggahan multibagian yang tidak lengkap.

     Apabila Anda membatasi cakupan aturan Siklus Hidup menggunakan tag objek, Anda tidak dapat memilih Hapus penanda hapus objek kedaluwarsa. Anda juga tidak dapat memilih Hapus penanda hapus objek kedaluwarsa jika Anda memilih Jadikan versi objek saat ini kedaluwarsa.

> **Note**
>
> Filter berbasis ukuran tidak dapat digunakan dengan penanda hapus dan unggahan multibagian yang tidak lengkap.

8. Apabila Anda memilih Jadikan versi objek saat ini kedaluwarsa atau Hapus versi objek tidak terkini secara permanen, konfigurasikan pemicu aturan berdasarkan tanggal tertentu atau umur objek.
9. Apabila Anda memilih Hapus penanda hapus objek kedaluwarsa, untuk mengonfirmasi bahwa Anda ingin menghapus penanda hapus objek kedaluwarsa, pilih Hapus penanda hapus objek kedaluwarsa.
10. Pilih Simpan.

Menghapus aturan siklus hidup

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih bucket Outposts yang ingin Anda hapus aturan siklus hidupnya.
4. Pilih tab Manajemen, kemudian dalam Aturan Siklus Hidup, pilih aturan yang ingin Anda hapus.
5. Pilih Hapus.

Membuat dan mengelola konfigurasi siklus hidup dengan menggunakan AWS CLI dan SDK for Java

Anda dapat menggunakan Siklus Hidup S3 untuk mengoptimalkan kapasitas penyimpanan Amazon S3 di Outposts. Anda dapat membuat aturan siklus hidup untuk menjadikan objek kedaluwarsa seiring bertambahnya umur objek atau digantikan oleh versi yang lebih baru. Anda dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk informasi selengkapnya tentang Siklus Hidup S3, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.

> **Note**
>
> Akun AWS Yang membuat bucket memilikinya dan merupakan satu-satunya yang dapat membuat, mengaktifkan, menonaktifkan, atau menghapus aturan siklus hidup.

Untuk membuat dan mengelola konfigurasi siklus hidup bucket S3 di Outposts dengan menggunakan AWS Command Line Interface (AWS CLI) dan file AWS SDK untuk Java, lihat contoh berikut.

Topik

MENEMPATKAN konfigurasi siklus hidup

AWS CLI

AWS CLI Contoh berikut menempatkan kebijakan konfigurasi siklus hidup pada bucket Outposts. Kebijakan ini menentukan bahwa semua objek yang ditandai dengan prefiks (*myprefix*) dan tag akan kedaluwarsa setelah 10 hari. Untuk menggunakan contoh ini, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

1. Simpan kebijakan konfigurasi siklus ke file JSON. Dalam contoh ini, file dinamai `lifecycle1.json`.

```
{
    "Rules": [
        {
            "ID": "id-1",
            "Filter": {
                "And": {
                    "Prefix": "myprefix",
                    "Tags": [
                        {
                            "Value": "mytagvalue1",
                            "Key": "mytagkey1"
                        },
                        {
                            "Value": "mytagvalue2",
                            "Key": "mytagkey2"
                        }
                    ],
                    "ObjectSizeGreaterThan": 1000,
                    "ObjectSizeLessThan": 5000
                }
            },
            "Status": "Enabled",
            "Expiration": {
                "Days": 10
            }
```

```
        }
    ]
}
```

2. Kirim file JSON sebagai bagian dari perintah CLI `put-bucket-lifecycle-configuration`. Untuk menggunakan perintah ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat put-bucket-lifecycle-configurationdi AWS CLI Referensi.

```
aws s3control put-bucket-lifecycle-configuration --account-id 123456789012 --
bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/
bucket/example-outposts-bucket --lifecycle-configuration file://lifecycle1.json
```

SDK for Java

Contoh SDK for Java berikut menempatkan konfigurasi siklus hidup pada bucket Outposts. Konfigurasi siklus hidup ini menentukan bahwa semua objek yang ditandai dengan prefiks (*`myprefix`*) dan tag akan kedaluwarsa setelah 10 hari. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Untuk informasi selengkapnya, lihat PutBucketLifecycleConfiguration dalam Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void putBucketLifecycleConfiguration(String bucketArn) {

    S3Tag tag1 = new S3Tag().withKey("mytagkey1").withValue("mytagkey1");
    S3Tag tag2 = new S3Tag().withKey("mytagkey2").withValue("mytagkey2");

    LifecycleRuleFilter lifecycleRuleFilter = new LifecycleRuleFilter()
            .withAnd(new LifecycleRuleAndOperator()
                    .withPrefix("myprefix")
                    .withTags(tag1, tag2))
                    .withObjectSizeGreaterThan(1000)
                    .withObjectSizeLessThan(5000);

    LifecycleExpiration lifecycleExpiration = new LifecycleExpiration()
            .withExpiredObjectDeleteMarker(false)
            .withDays(10);

    LifecycleRule lifecycleRule = new LifecycleRule()
```

```
            .withStatus("Enabled")
            .withFilter(lifecycleRuleFilter)
            .withExpiration(lifecycleExpiration)
            .withID("id-1");

    LifecycleConfiguration lifecycleConfiguration = new LifecycleConfiguration()
            .withRules(lifecycleRule);

    PutBucketLifecycleConfigurationRequest reqPutBucketLifecycle = new
 PutBucketLifecycleConfigurationRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn)
            .withLifecycleConfiguration(lifecycleConfiguration);

    PutBucketLifecycleConfigurationResult respPutBucketLifecycle =
 s3ControlClient.putBucketLifecycleConfiguration(reqPutBucketLifecycle);
    System.out.printf("PutBucketLifecycleConfiguration Response: %s%n",
 respPutBucketLifecycle.toString());

}
```

MENDAPATKAN konfigurasi siklus hidup di bucket S3 di Outposts

AWS CLI

AWS CLI Contoh berikut mendapatkan konfigurasi siklus hidup pada bucket Outposts. Untuk menggunakan perintah ini, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri. Untuk informasi selengkapnya tentang perintah ini, lihat get-bucket-lifecycle-configurationdi AWS CLI Referensi.

```
aws s3control get-bucket-lifecycle-configuration --account-id 123456789012 --bucket
 arn:aws:s3-outposts:<your-region>:123456789012:outpost/op-01ac5d28a6a232904/
bucket/example-outposts-bucket
```

SDK for Java

Contoh SDK for Java berikut mendapatkan konfigurasi siklus hidup pada bucket Outposts. Untuk informasi selengkapnya, lihat GetBucketLifecycleConfiguration di Referensi API Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;
```

```
public void getBucketLifecycleConfiguration(String bucketArn) {

    GetBucketLifecycleConfigurationRequest reqGetBucketLifecycle = new
 GetBucketLifecycleConfigurationRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    GetBucketLifecycleConfigurationResult respGetBucketLifecycle =
 s3ControlClient.getBucketLifecycleConfiguration(reqGetBucketLifecycle);
    System.out.printf("GetBucketLifecycleConfiguration Response: %s%n",
 respGetBucketLifecycle.toString());

}
```

Replikasi objek untuk S3 di Outposts.

Dengan Replikasi S3 aktif AWS Outposts, Anda dapat mengonfigurasi Amazon S3 di Outposts untuk secara otomatis mereplikasi objek S3 di Outposts yang berbeda, atau di antara bucket di Outpost yang sama. Anda dapat menggunakan Replikasi S3 di Outposts untuk memelihara beberapa replika data Anda di Outposts yang sama atau berbeda, atau di berbagai akun, untuk membantu memenuhi kebutuhan residensi data. Replikasi S3 di Outposts membantu memenuhi kebutuhan penyimpanan yang sesuai dan berbagi data di seluruh akun. Jika Anda perlu memastikan bahwa replika Anda identik dengan data sumber, Anda dapat menggunakan Replikasi S3 di Outposts untuk membuat replika objek Anda yang mempertahankan semua metadata, seperti waktu pembuatan objek asli, tag, dan versi. IDs

Replikasi S3 di Outposts juga menyediakan metrik dan notifikasi terperinci untuk memantau status replikasi objek antar bucket. Anda dapat menggunakan Amazon CloudWatch untuk memantau kemajuan replikasi dengan melacak byte replikasi yang tertunda, operasi menunggu replikasi, dan latensi replikasi antara bucket sumber dan tujuan Anda. Untuk mendiagnosis dan memperbaiki masalah konfigurasi dengan cepat, Anda juga dapat mengatur Amazon EventBridge untuk menerima pemberitahuan tentang kegagalan objek replikasi. Untuk mempelajari selengkapnya, lihat Mengelola replikasi Anda.

Topik

- Konfigurasi Replikasi
- Persyaratan Replikasi S3 di Outposts.

- Apa yang direplikasi?
- Apa yang tidak direplikasi?
- Apa yang tidak didukung oleh Replikasi S3 di Outposts?
- Menyiapkan replikasi
- Mengelola replikasi Anda

Konfigurasi Replikasi

S3 on Outposts. menyimpan konfigurasi replikasi sebagai XMLs. Dalam file XHTML konfigurasi replikasi, Anda menentukan peran AWS Identity and Access Management (IAM) dan satu atau beberapa aturan.

```
<ReplicationConfiguration>
    <Role>IAM-role-ARN</Role>
    <Rule>
        ...
    </Rule>
    <Rule>
         ...
    </Rule>
     ...
</ReplicationConfiguration>
```

S3 on Outposts. tidak dapat mereplikasi objek tanpa izin Anda. Anda memberikan izin S3 on Outposts. peran IAM yang Anda tentukan dalam konfigurasi replikasi. S3 on Outposts. mengasumsikan peran IAM untuk mereplikasi objek atas nama Anda. Anda harus memberikan izin yang diperlukan untuk peran IAM sebelum memulai replikasi. Untuk informasi selengkapnya tentang izin ini untuk S3 di Outposts. lihat. Membuat peran IAM

Anda menambahkan satu aturan dalam konfigurasi replikasi dalam skenario berikut:

- Anda ingin mereplikasi semua objek.
- Anda ingin mereplikasi satu subset objek. Anda mengidentifikasi subset objek dengan menambahkan filter pada aturan. Dalam filter, Anda menentukan awalan kunci objek, tag, atau kombinasi keduanya, untuk mengidentifikasi bagian objek yang diterapkan aturan.

Anda menambahkan beberapa aturan dalam konfigurasi replikasi jika Anda ingin mereplikasi subset objek yang berbeda. Dalam setiap aturan, Anda menetapkan filter yang memilih subset objek yang

berbeda. Misalnya, Anda dapat memilih untuk mereplikasi objek yang memiliki awalan kunci `tax/` atau `document/`. Untuk melakukan ini, Anda menambahkan dua aturan, satu yang menentukan filter awalan kunci `tax/` dan satu lagi yang menentukan awalan kunci `document/`.

Untuk informasi selengkapnya tentang S3 tentang konfigurasi replikasi Outposts dan aturan replikasi, lihat ReplicationConfiguration di Referensi API Amazon Simple Storage Service.

Persyaratan Replikasi S3 di Outposts.

Replikasi memerlukan hal berikut:

- Rentang CIDR Outpost tujuan harus dikaitkan dalam tabel subnet Outpost sumber Anda. Untuk informasi selengkapnya, lihat Prasyarat untuk membuat aturan replikasi.
- Bucket sumber dan tujuan harus memiliki Penentuan Versi S3 yang diaktifkan. Untuk informasi selengkapnya tentang penentuan versi, lihat Mengelola Penentuan Versi untuk bucket S3 di Outposts.
- Amazon S3 di Outposts. harus memiliki izin untuk mereplikasi objek dari bucket sumber ke bucket tujuan atas nama Anda. Itu berarti Anda harus membuat peran layanan untuk mendelegasikan `GET` dan `PUT` izin ke S3 di Outposts.
  1. Sebelum membuat peran layanan, Anda harus memiliki `GET` izin pada bucket sumber dan `PUT` izin di bucket tujuan.
  2. Untuk membuat peran layanan untuk mendelegasikan izin ke S3 di Outposts, Anda harus terlebih dahulu mengonfigurasi izin untuk mengizinkan entitas IAM (pengguna atau peran) untuk melakukan tindakan `iam:CreateRole` dan `iam:PassRole`. Kemudian, Anda mengizinkan entitas IAM untuk membuat peran layanan. Untuk membuat S3 pada Outpost mengasumsikan peran layanan atas nama Anda `GET` dan `PUT` mendelegasikan serta memberikan izin kepada S3 pada Outpost, Anda harus menetapkan kebijakan kepercayaan dan izin yang diperlukan untuk peran tersebut. Untuk informasi selengkapnya tentang izin ini untuk S3 di Outposts. lihat. Membuat peran IAM Untuk informasi selengkapnya tentang pembuatan peran layanan, lihat Membuat peran layanan.

Apa yang direplikasi?

Secara default, S3 on Outposts. mereplikasi berikut:

- Objek yang dibuat setelah Anda menambahkan konfigurasi replikasi.

- Metadata objek dari objek sumber ke replika. Untuk informasi tentang cara mereplikasi metadata dari replika ke objek sumber, lihat. Status replikasi jika sinkronisasi modifikasi replika Amazon S3 di Outposts diaktifkan
- Tag objek, jika ada.

Bagaimana cara menghapus operasi yang memengaruhi replikasi

Jika Anda menghapus objek dari bucket sumber, tindakan berikut terjadi secara default:

- Jika Anda membuat `DELETE` permintaan tanpa menentukan ID versi objek, S3 di Outposts. menambahkan penanda hapus. S3 on Outposts. berkaitan dengan penanda hapus sebagai berikut:
  - S3 di Outposts tidak mereplikasi penanda hapus secara default.
  - Namun, Anda dapat menambahkan replikasi penanda hapus ke non-tag-based aturan. Untuk informasi selengkapnya tentang cara mengaktifkan replikasi delete marker dalam konfigurasi replikasi Anda, lihat. Menggunakan konsol S3
- Jika Anda menentukan ID versi objek yang akan dihapus dalam `DELETE` permintaan, S3 on Outposts. menghapus versi objek tersebut dalam bucket sumber. Namun, itu tidak mereplikasi penghapusan di bucket tujuan. Dengan kata lain, itu tidak menghapus versi objek yang sama dari bucket tujuan. Cara ini melindungi data dari penghapusan yang berbahaya.

Apa yang tidak direplikasi?

Secara default, S3 di Outposts. tidak mereplikasi hal-hal berikut:

- Objek dalam bucket sumber yang merupakan replika yang dibuat oleh aturan replikasi lain. Misalnya, Anda mengonfigurasi replikasi dengan bucket A sebagai sumber dan bucket B sebagai tujuan. Sekarang bayangkan bahwa Anda menambahkan konfigurasi replikasi lain dengan bucket B sebagai sumber dan bucket C sebagai tujuan. Dalam hal ini, objek dalam bucket B yang merupakan replika objek dalam bucket A tidak direplikasi ke bucket C.
- Objek dalam bucket sumber yang telah direplikasi ke tujuan yang berbeda. Misalnya, jika Anda mengubah bucket tujuan dalam konfigurasi replikasi yang ada, S3 di Outposts. tidak akan mereplikasi objek tersebut lagi.
- Objek yang dibuat dengan enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C).

- Pembaruan untuk subsumber daya tingkat bucket.

  Misalnya, jika Anda mengubah konfigurasi siklus aktif atau menambahkan konfigurasi pemberitahuan ke bucket sumber Anda, perubahan ini tidak diterapkan ke bucket tujuan. Fitur ini memungkinkan untuk memiliki konfigurasi berbeda pada bucket sumber dan tujuan.

- Tindakan yang dilakukan berdasarkan konfigurasi siklus aktif.

  Misalnya, jika Anda mengaktifkan konfigurasi siklus aktif hanya pada bucket sumber Anda dan mengonfigurasi tindakan kedaluwarsa, S3 on Outposts. membuat delete marker untuk objek kedaluwarsa di bucket sumber tetapi tidak mereplikasi penanda tersebut ke bucket tujuan. Jika Anda ingin konfigurasi siklus aktif yang sama diterapkan pada bucket sumber dan tujuan, aktifkan konfigurasi siklus aktif yang sama pada keduanya. Untuk informasi lebih lanjut tentang konfigurasi siklus aktif, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.

Apa yang tidak didukung oleh Replikasi S3 di Outposts?

Fitur Replikasi S3 berikut saat ini tidak didukung oleh S3 di Outposts. berikut:

- Kontrol Waktu Replikasi S3 (S3 RTC). S3 RTC tidak didukung karena lalu lintas objek di Replikasi S3 di Outposts berjalan melalui jaringan lokal Anda (gateway lokal). Untuk informasi selengkapnya tentang gateway lokal, lihat Bekerja dengan gateway lokal di AWS Outposts Panduan Pengguna.
- Replikasi S3 untuk Operasi Batch.

Menyiapkan replikasi

> **Note**
>
> Objek yang sudah berada dalam bucket Anda sebelum pembuatan aturan replikasi tidak akan direplikasi secara otomatis. Dengan kata lain, Amazon S3 di Outposts tidak mereplikasi objek secara retroaktif. Untuk mereplikasi objek yang dibuat sebelum konfigurasi replikasi, Anda dapat menggunakan operasi API `CopyObject` untuk menyalinnya ke bucket yang sama. Setelah objek disalin, objek tersebut akan muncul sebagai objek "baru" di dalam bucket dan konfigurasi replikasi akan diterapkan pada objek tersebut. Untuk informasi selengkapnya tentang menyalin objek, lihat Menyalin objek di bucket Amazon S3 di Outposts

menggunakan AWS SDK untuk Java dan CopyObject di Referensi API Amazon Simple Storage Service.

Untuk mengaktifkan Replikasi S3 di Outposts, tambahkan aturan replikasi ke bucket Outposts sumber Anda. Aturan replikasi memerintahkan S3 pada Outpost untuk mereplikasi objek seperti yang ditentukan. Dalam aturan replikasi, Anda harus menyediakan:

- Titik akses bucket Outposts sumber–Titik akses Amazon Resource Name (ARN) atau alias titik akses dari bucket tempat Anda ingin S3 di Outposts untuk mereplikasi objek. Untuk informasi selengkapnya tentang penggunaan alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.
- Objek yang ingin Anda replikasi–Anda dapat mereplikasi semua objek di bucket sumber Outposts atau subset. Anda mengidentifikasi subset dengan menyediakan awalan nama kunci, satu atau lebih tag objek, atau keduanya dalam konfigurasi.

  Misalnya, jika Anda mengonfigurasi aturan replikasi untuk hanya mereplikasi objek dengan prefiks nama kunci `Tax/`, S3 pada Outposts akan mereplikasi objek dengan kunci seperti `Tax/doc1` atau `Tax/doc2`. Namun, aturan ini tidak mereplikasi objek dengan kunci `Legal/doc3`. Jika Anda menentukan prefiks dan satu atau beberapa tag, S3 pada Outpost hanya akan mereplikasi objek yang memiliki prefiks dan tag kunci tertentu.
- Bucket Outposts tujuan–ARN atau alias titik akses dari bucket tempat Anda ingin S3 di Outposts untuk mereplikasi objek.

Anda dapat mengonfigurasi aturan replikasi menggunakan REST API,, AWS Command Line Interface (AWS CLI) AWS SDKs, atau konsol Amazon S3.

S3 di Outposts juga menyediakan operasi API untuk mendukung pengaturan aturan replikasi. Untuk informasi selengkapnya, lihat topik berikut di Referensi API Amazon Simple Storage Service:

- PutBucketReplication
- GetBucketReplication
- DeleteBucketReplication

Topik

- Prasyarat untuk membuat aturan replikasi

- Membuat aturan replikasi di Outposts

Prasyarat untuk membuat aturan replikasi

Topik

- Menghubungkan subnet Outpost sumber dan tujuan
- Membuat peran IAM

Menghubungkan subnet Outpost sumber dan tujuan

Agar lalu lintas replikasi berjalan dari Outpost sumber ke Outpost tujuan melalui gateway lokal, Anda harus menambahkan rute baru untuk mengatur jaringan. Anda harus menghubungkan rentang jaringan Perutean Antar Domain Tanpa Kelas (CIDR) dari titik akses Anda secara bersamaan. Untuk setiap pasangan titik akses, Anda hanya perlu mengatur sambungan ini satu kali.

Beberapa langkah untuk menyiapkan koneksi akan berbeda-beda, tergantung pada jenis akses titik akhir Outpost yang terkait dengan titik akses Anda. Jenis akses untuk endpoint adalah Private (direct virtual private cloud [VPC] routing AWS Outposts for) atau IP milik Pelanggan (kumpulan alamat IP milik pelanggan [CoIP pool] dalam jaringan lokal Anda).

Langkah 1: Temukan rentang CIDR dari titik akhir Outposts sumber Anda

Untuk menemukan rentang CIDR dari titik akhir sumber yang terkait dengan titik akses sumber Anda

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Di daftar bucket Outposts, pilih bucket sumber yang Anda inginkan untuk replikasi.
4. Pilih tab Titik akses Outposts, dan pilih titik akses Outposts untuk bucket sumber untuk aturan replikasi Anda.
5. Pilih titik akhir Outposts.
6. Salin ID subnet yang akan digunakan pada Langkah 5.
7. Metode yang Anda gunakan untuk menemukan rentang CIDR dari titik akhir Outposts sumber bergantung pada jenis akses titik akhir Anda.

   Di bagian gambaran umum titik akhir Outposts, lihat Jenis Akses.

- Jika jenis aksesnya Privat, salin nilai Perutean Antar Domain Tanpa Kelas (CIDR) yang akan digunakan pada Langkah 6.
- Jika jenis aksesnya adalah IP Milik Pelanggan, lakukan hal berikut:
  1. Salin nilai IPv4 pool milik Pelanggan untuk digunakan sebagai ID dari kumpulan alamat nantinya.
  2. Buka AWS Outposts konsol di https://console.aws.amazon.com/outposts/.
  3. Di panel navigasi, pilih Tabel rute gateway lokal.
  4. Pilih nilai ID tabel rute gateway lokal dari Outpost sumber Anda.
  5. Di panel detail, pilih tab kolam CoIP. Tempelkan nilai ID kolam CoIP yang Anda salin sebelumnya di kotak pencarian.
  6. Untuk kumpulan CoIP yang cocok, salin CIDRsnilai yang sesuai dari titik akhir Outposts sumber Anda untuk digunakan pada Langkah 6.

Langkah 2: Temukan ID subnet dan rentang CIDR dari titik akhir Outposts tujuan Anda

Untuk menemukan ID subnet dan rentang CIDR dari titik akhir tujuan Anda yang terkait dengan titik akses tujuan, ikuti sublangkah yang sama di Langkah 1 dan ubah titik akhir Outposts sumber Anda ke titik akhir Outposts tujuan saat Anda menerapkan sublangkah tersebut. Salin nilai subnet ID dari titik akhir Outposts tujuan Anda yang akan digunakan pada Langkah 6. Salin nilai CIDR dari titik akhir Outposts tujuan Anda yang akan digunakan pada Langkah 5.

Langkah 3: Temukan ID gateway lokal dari Outpost sumber Anda

Untuk menemukan ID gateway lokal dari Outpost sumber Anda

1. Buka AWS Outposts konsol di https://console.aws.amazon.com/outposts/.
2. Di panel navigasi kiri, pilih Gateway lokal.
3. Pada halaman gateway Lokal, temukan ID Outpost dari Outpost sumber Anda yang ingin Anda gunakan untuk replikasi.
4. Salin nilai ID gateway lokal dari Outpost sumber Anda yang akan digunakan pada Langkah 5.

Untuk informasi selengkapnya tentang gateway lokal, lihat gateway Lokal di Panduan Pengguna AWS Outposts .

Langkah 4: Temukan ID gateway lokal dari Outpost tujuan Anda

Untuk menemukan ID gateway lokal Outpost tujuan Anda, ikuti sublangkah yang sama di Langkah 3, kecuali cari ID Outpost untuk Outpost tujuan Anda. Salin nilai ID gateway lokal dari Outpost tujuan Anda yang akan digunakan pada Langkah 6.

Langkah 5: Siapkan koneksi dari subnet Outpost sumber ke subnet Outpost tujuan Anda

Untuk menghubungkan dari subnet Outpost sumber ke subnet Outpost tujuan Anda

1. Masuk ke AWS Management Console dan buka konsol VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi kiri, pilih Subnets.
3. Di kotak pencarian, masukkan ID subnet untuk titik akhir Outposts sumber yang Anda dapatkan di Langkah 1. Pilih subnet dengan subnet ID yang cocok.
4. Untuk item subnet yang cocok, pilih Nilai Tabel rute dari subnet ini.
5. Pada halaman dengan tabel rute yang dipilih, pilih Tindakan, lalu pilih Edit rute.
6. Pada halaman Edit rute, pilih Tambahkan rute.
7. Di bawah Tujuan, masukkan rentang CIDR dari titik akhir Outposts tujuan Anda yang Anda dapatkan di Langkah 2.
8. Di bawah Target, pilih Gateway Lokal Outpost , dan masukkan ID gateway lokal dari Outpost sumber yang Anda dapatkan di Langkah 3.
9. Pilih Simpan perubahan.
10. Pastikan Status untuk rute tersebut Aktif.

Langkah 6: Siapkan koneksi dari subnet Outpost tujuan ke subnet Outpost sumber Anda

1. Masuk ke AWS Management Console dan buka konsol VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi kiri, pilih Subnets.
3. Di kotak pencarian, masukkan ID subnet untuk titik akhir Outposts tujuan yang Anda dapatkan di Langkah 2. Pilih subnet dengan subnet ID yang cocok.
4. Untuk item subnet yang cocok, pilih Nilai Tabel rute dari subnet ini.
5. Pada halaman dengan tabel rute yang dipilih, pilih Tindakan, lalu pilih Edit rute.
6. Pada halaman Edit rute, pilih Tambahkan rute.

7. Di bawah Tujuan, masukkan rentang CIDR dari titik akhir Outposts sumber Anda yang Anda dapatkan di Langkah 1.
8. Di bawah Target, pilih Gateway Lokal Outpost, dan masukkan ID gateway lokal dari Outpost tujuan yang Anda temukan di Langkah 4.
9. Pilih Simpan perubahan.
10. Pastikan Status untuk rute tersebut Aktif.

Setelah Anda menghubungkan rentang jaringan CIDR dari titik akses sumber dan tujuan Anda, Anda harus membuat peran AWS Identity and Access Management (IAM).

Membuat peran IAM

Secara default, semua sumber daya S3 di Outposts—bucket, objek, dan subsumber terkait—bersifat privat, dan hanya pemilik sumber daya yang bisa mengakses sumber daya. S3 di Outposts membutuhkan izin untuk membaca dan mereplikasi objek dari bucket Outposts. Anda memberikan izin ini dengan membuat peran layanan IAM dan menentukan peran tersebut dalam konfigurasi replikasi.

Bagian ini menjelaskan kebijakan kepercayaan dan kebijakan izin minimum yang diperlukan. Contoh panduan memberikan step-by-step instruksi untuk membuat peran IAM. Untuk informasi selengkapnya, lihat Membuat aturan replikasi di Outposts. Untuk informasi selengkapnya tentang peran IAM, lihat Peran IAM dalam Panduan Pengguna IAM.

- Contoh berikut menunjukkan kebijakan kepercayaan, yaitu S3 di Outposts diidentifikasi sebagai pengguna utama layanan yang dapat menjalankan peran tersebut.

  JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"s3-outposts.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
```

```
}
```

- Contoh berikut menunjukkan kebijakan akses, yaitu ketika Anda memberikan izin peran untuk melakukan tugas replikasi atas nama Anda. Saat S3 di Outposts mengambil peran tersebut, S3 memiliki izin yang Anda tentukan dalam kebijakan ini. Untuk menggunakan kebijakan ini, ganti *`user input placeholders`* dengan informasi Anda sendiri. Pastikan untuk menggantinya dengan Pos Luar IDs Outposts sumber dan tujuan Anda serta nama bucket dan nama titik akses dari bucket Outposts sumber dan tujuan Anda.

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:GetObjectVersionForReplication",
                "s3-outposts:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:123456789012:outpost/SOURCE-
OUTPOST-ID/bucket/SOURCE-OUTPOSTS-BUCKET/object/*",
                "arn:aws:s3-outposts:us-east-1:123456789012:outpost/SOURCE-
OUTPOST-ID/accesspoint/SOURCE-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:ReplicateObject",
                "s3-outposts:ReplicateDelete"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/bucket/DESTINATION-OUTPOSTS-
BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/accesspoint/DESTINATION-
OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
```

```
}
```

Kebijakan akses memberikan izin untuk tindakan berikut:

- `s3-outposts:GetObjectVersionForReplication`–Izin untuk tindakan ini diberikan pada semua objek sehingga memungkinkan S3 pada Outpost untuk mendapatkan versi objek tertentu yang terkait dengan setiap objek.

- `s3-outposts:GetObjectVersionTagging`–Izin untuk tindakan ini pada objek di bucket *`SOURCE-OUTPOSTS-BUCKET`* (bucket sumber) memungkinkan S3 di Outposts membaca tanda objek untuk replikasi. Untuk informasi selengkapnya, lihat Menambahkan tag untuk bucket S3 pada Outposts. Jika S3 pada Outpost tidak memiliki izin ini, maka S3 akan mereplikasi objek, tetapi tidak mereplikasi tanda objek.
- `s3-outposts:ReplicateObject` dan `s3-outposts:ReplicateDelete`–Izin untuk tindakan ini pada semua objek di bucket *`DESTINATION-OUTPOSTS-BUCKET`* (bucket tujuan) memungkinkan S3 di Outposts mereplikasi objek atau penanda hapus ke bucket Outposts tujuan. Untuk informasi tentang penanda hapus, lihat Bagaimana cara menghapus operasi yang memengaruhi replikasi.

> **Note**
>
> - Izin untuk tindakan `s3-outposts:ReplicateObject` pada bucket *`DESTINATION-OUTPOSTS-BUCKET`* (bucket tujuan) juga memungkinkan replikasi tag objek. Oleh karena itu, Anda tidak perlu secara eksplisit memberikan izin untuk tindakan `s3-outposts:ReplicateTags` tersebut.
> - Untuk replikasi lintas akun, pemilik bucket Outposts tujuan harus memperbarui kebijakan bucket guna memberikan izin untuk tindakan `s3-outposts:ReplicateObject` pada *`DESTINATION-OUTPOSTS-BUCKET`*. `s3-outposts:ReplicateObject`Tindakan ini memungkinkan S3 di Outposts mereplikasi objek dan tag objek ke bucket Outposts tujuan.

Untuk daftar tindakan S3 tentang Outposts, lihat Tindakan yang ditentukan oleh S3 di Outpost.

> **Important**
>
> Akun AWS Yang memiliki peran IAM harus memiliki izin untuk tindakan yang diberikannya ke peran IAM.

Sebagai contoh, bucket Outposts sumber berisi objek yang dimiliki oleh Akun AWS lain. Pemilik objek harus secara eksplisit memberikan izin Akun AWS yang diperlukan kepada pemilik peran IAM melalui kebijakan bucket dan kebijakan jalur akses. Jika tidak, S3 di Outposts tidak dapat mengakses objek, dan replikasi objek gagal.
Izin yang dijelaskan di sini terkait dengan konfigurasi replikasi minimum. Jika Anda memilih untuk menambahkan konfigurasi replikasi opsional, Anda harus memberikan izin tambahan ke S3 di Outposts.

Memberikan izin saat bucket Outposts sumber dan tujuan dimiliki oleh yang berbeda Akun AWS

Ketika bucket Outposts sumber dan tujuan tidak dimiliki oleh akun yang sama, pemilik bucket Outposts tujuan harus memperbarui kebijakan bucket dan titik akses untuk bucket tujuan tersebut. Kebijakan ini harus mengizinkan pemilik bucket Outposts sumber dan peran layanan IAM untuk melakukan tindakan replikasi, seperti yang ditunjukkan pada contoh kebijakan berikut ini, jika tidak, replikasi akan gagal. Dalam contoh kebijakan ini, *DESTINATION-OUTPOSTS-BUCKET* adalah bucket tujuan. Untuk menggunakan contoh kebijakan ini, ganti *user input placeholders* dengan informasi Anda sendiri.

Jika Anda membuat peran layanan IAM secara manual, tetapkan jalur peran sebagai `role/service-role/`, seperti yang ditunjukkan dalam contoh kebijakan berikut. Untuk informasi selengkapnya, lihat IAM ARNs di Panduan Pengguna IAM.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForDestinationBucket",
    "Statement": [
        {
            "Sid": "Permissions on objects",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/service-role/source-
account-IAM-role"
            },
            "Action": [
                "s3-outposts:ReplicateDelete",
                "s3-outposts:ReplicateObject"
            ],
```

```
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:444455556666:outpost/DESTINATION-
OUTPOST-ID/bucket/DESTINATION-OUTPOSTS-BUCKET/object/*"
            ]
        }
    ]
}
```

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForDestinationAccessPoint",
    "Statement": [
        {
            "Sid": "Permissions on objects",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/service-role/source-
account-IAM-role"
            },
            "Action": [
                "s3-outposts:ReplicateDelete",
                "s3-outposts:ReplicateObject"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:111122223333:outpost/DESTINATION-
OUTPOST-ID/accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
}
```

> **Note**
>
> Jika objek dalam bucket Outposts ditandai, perhatikan hal berikut:
> Jika pemilik bucket Outposts memberikan izin S3 di Outposts untuk tindakan `s3-outposts:GetObjectVersionTagging` dan `s3-outposts:ReplicateTags` agar dapat

mereplikasi tanda objek (melalui peran IAM), Amazon S3 mereplikasi tanda beserta objek tersebut. Untuk informasi tentang peran IAM, lihat Membuat peran IAM.

Membuat aturan replikasi di Outposts

Replikasi S3 di Outposts adalah replikasi objek otomatis dan asinkron di seluruh ember dalam hal yang sama atau berbeda. AWS Outposts Replikasi menyalin objek yang baru dibuat dan pembaruan objek dari bucket Outposts ke bucket atau bucket Outposts tujuan. Untuk informasi selengkapnya, lihat Replikasi objek untuk S3 di Outposts..

> **Note**
>
> Objek yang ada di bucket Outposts sumber sebelum Anda menyiapkan aturan replikasi tidak akan direplikasi. Dengan kata lain, S3 di Outposts tidak mereplikasi objek secara retroaktif. Untuk mereplikasi objek yang dibuat sebelum konfigurasi replikasi, Anda dapat menggunakan operasi API `CopyObject` untuk menyalinnya ke bucket yang sama. Setelah objek disalin, objek tersebut akan muncul sebagai objek "baru" di dalam bucket dan konfigurasi replikasi akan diterapkan pada objek tersebut. Untuk informasi selengkapnya tentang menyalin objek, lihat Menyalin objek di bucket Amazon S3 di Outposts menggunakan AWS SDK untuk Java dan CopyObject di Referensi API Amazon Simple Storage Service.

Saat mengonfigurasi replikasi, Anda perlu menambahkan aturan replikasi ke bucket Outposts. Aturan replikasi menentukan objek bucket Outposts yang akan direplikasi dan bucket Outposts tujuan atau bucket tempat objek yang telah direplikasi akan disimpan. Anda dapat membuat aturan untuk mereplikasi semua objek dalam bucket atau subset objek dengan awalan nama kunci tertentu, satu atau beberapa tag objek, atau keduanya. Bucket Outposts tujuan dapat berada di Outpost yang sama dengan bucket Outposts sumber, atau bisa juga di Outpost yang berbeda.

Untuk aturan replikasi S3 di Outposts, Anda harus menyediakan titik akses Amazon Resource Name (ARN) bucket Outposts sumber dan titik akses ARN bucket Outposts tujuan, bukan nama bucket Outposts sumber dan tujuan.

Jika Anda menentukan ID versi objek yang akan dihapus, S3 di Outposts akan menghapus versi objek tersebut di bucket Outposts sumber. Tapi tindakan itu tidak akan mereplikasi penghapusan ke bucket Outposts. Dengan kata lain, tindakan ini tidak akan menghapus versi objek yang sama dari bucket Outposts. Cara ini melindungi data dari penghapusan yang berbahaya.

Saat Anda menambahkan aturan replikasi ke bucket Outposts, aturan tersebut diaktifkan secara default, sehingga aturan mulai bekerja segera setelah Anda menyimpannya.

Dalam contoh ini, Anda menyiapkan replikasi untuk bucket Outposts sumber dan tujuan yang berada di Outposts yang berbeda dan dimiliki oleh Akun AWS yang sama. Contoh disediakan untuk menggunakan konsol Amazon S3, AWS Command Line Interface (AWS CLI), dan dan. AWS SDK untuk Java AWS SDK untuk .NET Untuk informasi tentang izin Replikasi S3 lintas akun di Outpost, lihat Memberikan izin saat bucket Outposts sumber dan tujuan dimiliki oleh yang berbeda Akun AWS.

Untuk prasyarat penyiapan aturan replikasi S3 di Outposts, lihat Prasyarat untuk membuat aturan replikasi.

Menggunakan konsol S3

Ikuti langkah-langkah ini untuk mengonfigurasi aturan replikasi saat bucket Amazon S3 di Outposts berada di Outpost yang berbeda dari bucket Outposts sumber.

Jika bucket Outposts berada di akun yang berbeda dari bucket Outposts sumber, Anda harus menambahkan kebijakan bucket ke bucket Outposts tujuan untuk memberi pemilik izin akun bucket Outposts sumber untuk mereplikasi objek di bucket Outposts tujuan.

Untuk membuat aturan replikasi

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di daftar Bucket Outposts, pilih nama bucket yang ingin Anda gunakan sebagai bucket sumber.
3. Pilih tab Manajemen, gulir ke bawah ke bagian Aturan replikasi, lalu pilih Buat aturan replikasi.
4. Untuk Nama aturan replikasi, masukkan nama untuk aturan Anda yang nantinya akan membantu mengidentifikasi aturan. Nama wajib diisi dan harus unik dalam bucket.
5. Di bawah Status, Diaktifkan dipilih secara default. Aturan yang diaktifkan mulai berfungsi segera setelah Anda menyimpannya. Jika Anda ingin mengaktifkan aturan nanti, pilih Dinonaktifkan.
6. Di bawah Prioritas, nilai prioritas aturan menentukan aturan mana yang akan diterapkan jika terdapat aturan yang tumpang tindih. Saat objek dimasukkan dalam cakupan lebih dari satu aturan replikasi, S3 di Outposts menggunakan nilai prioritas ini untuk menghindari konflik. Secara default, aturan baru akan ditambahkan ke konfigurasi replikasi pada prioritas tertinggi. Semakin tinggi angkanya, semakin tinggi prioritasnya.

   Untuk mengubah prioritas aturan, setelah Anda menyimpannya, pilih nama aturan dari daftar aturan replikasi, pilih Tindakan, lalu pilih Edit prioritas.

7. Di Bawah bucket sumber, Anda memiliki opsi berikut untuk mengatur sumber replikasi:

   - Untuk mereplikasi seluruh bucket, pilih Terapkan ke semua objek di bucket.
   - Untuk menerapkan prefiks atau pemfilteran tanda ke sumber replikasi, pilih Batasi cakupan aturan ini dengan menggunakan satu atau beberapa filter. Anda dapat menggabungkan awalan dan tag.
     - Untuk mereplikasi semua objek yang memiliki prefiks yang sama, di bawah Prefiks, masukkan prefiks di dalam kotak. Menggunakan filter Prefiks membatasi replikasi ke semua objek yang memiliki nama yang diawali dengan string yang sama (misalnya, `pictures`).

       Jika memasukkan prefiks yang merupakan nama folder, Anda harus menggunakan / (garis miring) sebagai karakter terakhir (misalnya, `pictures/`).
     - Untuk mereplikasi semua objek yang memiliki satu atau lebih tanda objek yang sama, pilih Tambahkan tanda, lalu masukkan pasangan nilai kunci ke dalam kotak. Untuk menambahkan tag lain, ulangi prosedur. Untuk informasi selengkapnya tentang tag objek, lihat Menambahkan tag untuk bucket S3 pada Outposts.
8. Untuk mengakses S3 di bucket sumber Outposts untuk replikasi, di bawah nama titik akses Sumber, pilih titik akses yang dilampirkan ke bucket sumber.
9. Di bawah Tujuan, pilih titik akses ARN dari bucket Outposts tujuan tempat Anda ingin S3 di Outposts untuk mereplikasi objek. Bucket Outposts tujuan bisa sama atau berbeda Akun AWS dengan bucket Outposts sumber.

   Jika bucket tujuan berada di akun yang berbeda dari bucket Outposts, Anda harus menambahkan kebijakan bucket ke bucket tujuan untuk memberikan izin kepada pemilik akun bucket Outposts sumber untuk mereplikasi objek ke bucket Outposts tujuan. Untuk informasi selengkapnya, lihat Memberikan izin saat bucket Outposts sumber dan tujuan dimiliki oleh yang berbeda Akun AWS.

   > **Note**
   >
   > Jika Penentuan Versi tidak diaktifkan pada bucket Outposts tujuan, Anda akan mendapatkan peringatan yang berisi tombol Aktifkan Penentuan Versi. Pilih tombol ini untuk mengaktifkan Penentuan Versi pada bucket.
10. Siapkan peran layanan AWS Identity and Access Management (IAM) yang dapat diasumsikan oleh S3 di Outposts untuk mereplikasi objek atas nama Anda.

Untuk mengatur peran IAM, di bawah peran IAM, lakukan salah satu dari yang berikut:

- Agar S3 di Outposts membuat peran IAM baru untuk konfigurasi replikasi Anda, pilih Pilih dari peran IAM yang ada lalu pilih Buat peran baru. Saat Anda menyimpan aturan, kebijakan baru akan dibuat untuk peran IAM yang sesuai dengan bucket Outposts sumber dan tujuan yang Anda pilih. Sebaiknya Anda memilih Buat peran baru.
- Anda juga dapat memilih untuk menggunakan peran IAM yang sudah ada. Jika memilih opsi tersebut, Anda harus memilih peran yang memberikan izin yang diperlukan untuk replikasi kepada S3 di Outpost. Jika peran ini tidak memberikan izin yang cukup kepada S3 di Outpost untuk mengikuti aturan replikasi Anda, replikasi akan gagal.

  Untuk memilih peran yang ada, pilih Pilih dari peran IAM yang ada, lalu pilih peran dari menu tarik-turun. Anda juga dapat memilih Masukkan peran IAM ARN dan kemudian masukkan Amazon Resource Name (ARN) peran IAM.

> **Important**
>
> Saat Anda menambahkan aturan replikasi ke bucket S3 di Outposts, Anda harus memiliki izin `iam:CreateRole` dan `iam:PassRole` untuk dapat membuat dan meneruskan peran IAM yang memberikan izin replikasi S3 di Outposts. Untuk informasi selengkapnya, lihat Memberikan izin pengguna untuk meneruskan peran ke Layanan AWS di Panduan Pengguna IAM.

11. Semua objek dalam bucket Outposts dienkripsi secara default. Untuk informasi selengkapnya tentang S3 tentang enkripsi Outposts, lihat Enkripsi data dalam S3 di Outposts. Hanya objek yang dienkripsi dengan enkripsi di sisi server dengan kunci yang terkelola Amazon S3 (SSE-S3) yang dapat direplikasi. Replikasi objek yang dienkripsi dengan enkripsi di sisi server dengan kunci AWS Key Management Service (AWS KMS) (SSE-KMS) atau enkripsi di sisi server dengan kunci enkripsi yang disediakan pelanggan (SSE-C) tidak didukung.
12. Bila perlu, aktifkan opsi tambahan berikut ini saat mengatur konfigurasi aturan replikasi:

    - Jika Anda ingin mengaktifkan metrik replikasi S3 pada Outposts dalam konfigurasi replikasi, pilih Metrik replikasi. Untuk informasi selengkapnya, lihat Memantau progres dengan metrik replikasi.

- Jika Anda ingin mengaktifkan replikasi penanda hapus dalam konfigurasi replikasi Anda, pilih Replikasi penanda hapus. Untuk informasi selengkapnya, lihat Bagaimana cara menghapus operasi yang memengaruhi replikasi.
- Jika Anda ingin mereplikasi perubahan metadata yang dibuat pada replika kembali ke objek sumber, pilih Sinkronisasi modifikasi replika. Untuk informasi selengkapnya, lihat Status replikasi jika sinkronisasi modifikasi replika Amazon S3 di Outposts diaktifkan.

13. Untuk menyelesaikannya, pilih Buat aturan.

Setelah menyimpan aturan, Anda dapat mengedit, mengaktifkan, menonaktifkan, atau menghapus aturan. Caranya, buka tab Manajemen untuk bucket Outposts sumber, gulir ke bawah ke bagian Aturan replikasi, pilih aturan Anda, lalu pilih Edit aturan.

Menggunakan AWS CLI

Untuk menggunakan AWS CLI untuk mengatur replikasi ketika bucket Outposts sumber dan tujuan dimiliki oleh yang Akun AWS sama, Anda melakukan hal berikut:

- Buat bucket Outposts sumber dan tujuan.
- Aktifkan Penentuan Versi pada kedua bucket.
- Buat peran IAM yang memberikan izin kepada S3 di Outpost untuk mereplikasi objek.
- Tambahkan konfigurasi replikasi ke bucket Outposts.

Untuk memverifikasi konfigurasi, Anda mengujinya.

Untuk mengatur replikasi ketika bucket Outposts sumber dan tujuan dimiliki oleh yang sama Akun AWS

1. Atur profil kredensial untuk AWS CLI. Dalam contoh ini, kami menggunakan nama profil `acctA`. Untuk informasi tentang mengatur profil kredensial, lihat Profil Bernama di Panduan Pengguna AWS Command Line Interface .

> **Important**
>
> Profil yang Anda gunakan untuk latihan ini harus memiliki izin yang diperlukan. Misalnya, dalam konfigurasi replikasi, Anda menentukan peran layanan IAM yang dapat digunakan S3 di Outposts. Anda dapat melakukan ini hanya jika profil yang Anda gunakan memiliki izin `iam:CreateRole` dan `iam:PassRole`. Untuk informasi selengkapnya, lihat

Memberikan izin pengguna untuk meneruskan peran ke Layanan AWS di Panduan Pengguna IAM. Jika Anda menggunakan kredensial administrator untuk membuat profil yang diberi nama, profil tersebut akan memiliki izin yang diperlukan untuk melakukan semua tugas.

2. Buat bucket dan aktifkan penentuan versi di dalamnya. Perintah `create-bucket` berikut akan membuat bucket *SOURCE-OUTPOSTS-BUCKET* di Wilayah AS Timur (Virginia Utara) (`us-east-1`). Untuk menggunakan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control create-bucket --bucket SOURCE-OUTPOSTS-BUCKET --outpost-id SOURCE-OUTPOST-ID --profile acctA --region us-east-1
```

Perintah `put-bucket-versioning` berikut memungkinkan Penentuan Versi pada bucket *SOURCE-OUTPOSTS-BUCKET*. Untuk menggunakan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-ID/bucket/SOURCE-OUTPOSTS-BUCKET --versioning-configuration Status=Enabled --profile acctA
```

3. Buat bucket tujuan dan aktifkan penentuan versi di dalamnya. Perintah `create-bucket` berikut akan membuat *DESTINATION-OUTPOSTS-BUCKET* bucket di Wilayah AS Barat (Oregon) (`us-west-2`). Untuk menggunakan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

> **Note**
>
> Untuk menyiapkan konfigurasi replikasi saat bucket Outposts sumber dan tujuan berada dalam kondisi yang Akun AWS sama, Anda menggunakan profil bernama yang sama. Contoh ini menggunakan `acctA`. Untuk menguji konfigurasi replikasi ketika bucket dimiliki oleh yang berbeda Akun AWS, Anda menentukan profil yang berbeda untuk setiap bucket.

```
aws s3control create-bucket --bucket DESTINATION-OUTPOSTS-BUCKET --create-bucket-
configuration LocationConstraint=us-west-2 --outpost-id DESTINATION-OUTPOST-ID --
profile acctA --region us-west-2
```

Perintah `put-bucket-versioning` berikut memungkinkan Penentuan Versi pada bucket *DESTINATION-OUTPOSTS-BUCKET*. Untuk menggunakan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/DESTINATION-OUTPOST-ID/bucket/DESTINATION-
OUTPOSTS-BUCKET --versioning-configuration Status=Enabled --profile acctA
```

4. Buat peran layanan IAM. Kemudian dalam konfigurasi replikasi, Anda menambahkan peran layanan ini ke bucket *SOURCE-OUTPOSTS-BUCKET*. S3 di Outposts menggunakan peran ini untuk mereplikasi objek atas nama Anda. Anda membuat peran IAM dalam dua langkah:

   a. Buat peran IAM.

      i. Salin kebijakan kepercayaan berikut dan simpan di file dengan nama `s3-on-outposts-role-trust-policy.json` di direktori saat ini di komputer lokal Anda. Kebijakan ini memberikan izin kepada pengguna utama layanan S3 di Outpost untuk menjalankan peran layanan tersebut.

      JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"s3-outposts.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

ii. Jalankan perintah berikut untuk membuat peran. Ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws iam create-role --role-name replicationRole --assume-role-policy-
document file://s3-on-outposts-role-trust-policy.json --profile acctA
```

b. Lampirkan kebijakan izin pada peran layanan.

i. Salin kebijakan izin berikut dan simpan ke file dengan nama `s3-on-outposts-role-permissions-policy.json` di direktori saat ini di komputer lokal Anda. Kebijakan ini memberikan izin untuk berbagai S3 di bucket Outposts dan tindakan objek. Untuk menggunakan kebijakan ini, ganti *user input placeholders* dengan informasi Anda sendiri.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:GetObjectVersionForReplication",
                "s3-outposts:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/SOURCE-OUTPOST-ID/bucket/SOURCE-OUTPOSTS-
BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/SOURCE-OUTPOST-ID/accesspoint/SOURCE-
OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:ReplicateObject",
                "s3-outposts:ReplicateDelete"
            ],
            "Resource": [
```

```
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/
bucket/DESTINATION-OUTPOSTS-BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/
accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
}
```

ii. Jalankan perintah berikut untuk membuat kebijakan dan melampirkannya ke peran. Ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws iam put-role-policy --role-name replicationRole --policy-
document file://s3-on-outposts-role-permissions-policy.json --policy-
name replicationRolePolicy --profile acctA
```

5. Menambahkan konfigurasi replikasi ke bucket *SOURCE-OUTPOSTS-BUCKET*.

a. Meskipun S3 pada Outposts API memerlukan konfigurasi replikasi dalam format XHTML, AWS CLI Anda harus menentukan konfigurasi replikasi dalam format JSON. Simpan JSON berikut dalam file yang disebut `replication.json` ke direktori lokal di komputer Anda. Untuk menggunakan konfigurasi ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
{
  "Role": "IAM-role-ARN",
  "Rules": [
    {
      "Status": "Enabled",
      "Priority": 1,
      "DeleteMarkerReplication": { "Status": "Disabled" },
      "Filter" : { "Prefix": "Tax"},
      "Destination": {
        "Bucket":
        "arn:aws:s3-outposts:region:123456789012:outpost/DESTINATION-OUTPOST-
ID/accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT"
      }
    }
  ]
}
```

b. Jalankan perintah `put-bucket-replication` berikut untuk menambahkan konfigurasi replikasi ke bucket Outposts Anda. Untuk menggunakan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control put-bucket-replication --account-id 123456789012 --
bucket arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-
ID/bucket/SOURCE-OUTPOSTS-BUCKET --replication-configuration file://
replication.json --profile acctA
```

c. Untuk mengambil konfigurasi replikasi, gunakan perintah `get-bucket-replication`. Untuk menggunakan perintah ini, ganti *`user input placeholders`* dengan informasi Anda sendiri.

```
aws s3control get-bucket-replication --account-id 123456789012 --bucket
 arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-ID/
bucket/SOURCE-OUTPOSTS-BUCKET --profile acctA
```

6. Uji pengaturan di konsol Amazon S3:

a. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/

b. Di bucket *`SOURCE-OUTPOSTS-BUCKET`*, buat folder bernama `Tax`.

c. Tambahkan objek sampel ke folder `Tax` di bucket *`SOURCE-OUTPOSTS-BUCKET`*.

d. Di bucket *`DESTINATION-OUTPOSTS-BUCKET`*, verifikasi hal berikut:

- S3 di Outpost mereplikasi objek tersebut.

> **Note**
>
> Durasi waktu yang dibutuhkan S3 di Outpost untuk mereplikasi sebuah objek bergantung pada ukuran objek tersebut. Untuk informasi tentang cara melihat status replikasi, lihat Mendapatkan informasi status replikasi.

- Pada tab Properti objek, Status replikasi diatur ke Replika (mengidentifikasi ini sebagai objek replika).

Mengelola replikasi Anda

Bagian ini menjelaskan opsi konfigurasi replikasi tambahan yang tersedia di S3 on Outposts, cara menentukan status replikasi, dan cara memecahkan masalah replikasi. Untuk informasi tentang konfigurasi replikasi inti, lihat Menyiapkan replikasi.

Topik

Memantau progres dengan metrik replikasi

Replikasi S3 di Outposts memberikan metrik terperinci untuk aturan replikasi dalam konfigurasi replikasi Anda. Dengan metrik replikasi, Anda dapat memantau progres replikasi dalam interval 5 menit dengan melacak byte yang menunggu replikasi, replikasi latensi replikasi, dan operasi tertunda. Untuk membantu memecahkan masalah konfigurasi apa pun, Anda juga dapat mengatur Amazon EventBridge untuk menerima pemberitahuan tentang kegagalan replikasi.

Saat metrik replikasi diaktifkan, Replikasi S3 di Outposts menerbitkan metrik berikut ke Amazon: CloudWatch

- Byte dengan Replikasi Tertunda–Jumlah total byte objek yang menunggu replikasi untuk aturan replikasi yang ditentukan.
- Latensi Replikasi–Jumlah detik maksimum saat bucket tujuan replikasi berada di belakang bucket sumber untuk aturan replikasi tertentu.
- Operasi Replikasi Tertunda–Jumlah operasi yang menunggu replikasi untuk aturan replikasi yang ditentukan. Operasi mencakup objek, penanda hapus, dan tag.

> **Note**
>
> Metrik Replikasi S3 pada Outposts ditagih dengan tarif yang sama dengan metrik kustom. CloudWatch Untuk informasi selengkapnya, lihat harga CloudWatch.

Mendapatkan informasi status replikasi

Status replikasi dapat membantu Anda mengetahui status objek saat ini yang sedang direplikasi oleh Amazon S3 on Outposts. Status replikasi dari objek sumber akan kembali sebagai `PENDING`, `COMPLETED`, atau `FAILED`. Status replikasi dari replika akan mengembalikan `REPLICA`.

Gambaran status replikasi

Dalam skenario replikasi, Anda memiliki bucket sumber tempat Anda mengonfigurasi replikasi dan bucket tujuan tempat S3 on Outposts mereplikasi objek. Saat Anda meminta sebuah objek (menggunakan `GetObject`) atau metadata objek (menggunakan `HeadObject`) dari bucket ini, S3 on Outposts mengembalikan header `x-amz-replication-status` dalam respons sebagai berikut:

- Saat Anda meminta objek dari bucket sumber, S3 on Outposts mengembalikan header `x-amz-replication-status` jika objek di permintaan Anda memenuhi syarat untuk replikasi.

  Misalnya, Anda menetapkan awalan objek `TaxDocs` dalam konfigurasi replikasi Anda untuk memberi tahu S3 on Outposts agar hanya mereplikasi objek dengan awalan nama kunci `TaxDocs`. Objek apa pun yang Anda unggah yang memiliki awalan nama kunci ini—misalnya, `TaxDocs/document1.pdf`—akan direplikasi. Untuk permintaan objek dengan awalan nama kunci ini, S3 on Outposts mengembalikan header `x-amz-replication-status` dengan salah satu nilai berikut untuk status replikasi objek: `PENDING`, `COMPLETED`, atau `FAILED`.

  > **Note**
  >
  > Jika replikasi objek gagal setelah Anda mengunggah sebuah objek, Anda tidak dapat mencoba ulang replikasi. Anda harus mengunggah objek lagi. Transisi objek ke status `FAILED` untuk masalah seperti tidak adanya izin peran replikasi atau izin bucket. Untuk kegagalan sementara, seperti jika bucket atau Outposts tidak tersedia, status replikasi tidak akan berpindah ke `FAILED`, tetapi tetap pada `PENDING`. Setelah sumber daya kembali online, S3 on Outposts melanjutkan replikasi objek tersebut.

- Saat Anda meminta objek dari bucket tujuan, jika objek dalam permintaan Anda adalah replika yang dibuat S3 on Outposts, S3 on Outposts mengembalikan header `x-amz-replication-status` dengan nilai `REPLICA`.

> **Note**
>
> Sebelum menghapus sebuah objek dari bucket sumber dengan replikasi yang diaktifkan, periksa status replikasi objek untuk memastikan bahwa objek tersebut telah direplikasi.

Status replikasi jika sinkronisasi modifikasi replika Amazon S3 di Outposts diaktifkan

Saat aturan replikasi Anda mengaktifkan sinkronisasi modifikasi replika S3 on Outposts, replika dapat melaporkan status selain `REPLICA`. Jika perubahan metadata sedang dalam proses replikasi, header `x-amz-replication-status` untuk replika mengembalikan `PENDING`. Jika sinkronisasi modifikasi replika gagal mereplikasi metadata, header untuk replika akan mengembalikan `FAILED`. Jika metadata direplikasi dengan benar, header untuk replika mengembalikan nilai `REPLICA`.

Memecahkan masalah replikasi

Jika replika objek tidak muncul di bucket Amazon S3 on Outposts tujuan setelah Anda mengonfigurasi replikasi, gunakan kiat pemecahan masalah ini untuk mengidentifikasi dan memperbaiki masalah.

- Waktu yang dibutuhkan S3 on Outposts untuk mereplikasi sebuah objek tergantung pada beberapa faktor, termasuk jarak antara Outposts sumber dan tujuan, serta ukuran objek.

  Anda dapat memeriksa status replikasi objek sumber. Jika status replikasi objek adalah `PENDING`, S3 on Outposts belum menyelesaikan replikasi. Jika status replikasi objek adalah `FAILED`, periksa konfigurasi replikasi yang ditetapkan pada bucket sumber.
- Dalam konfigurasi replikasi pada bucket sumber, verifikasi hal berikut:
  - Amazon Resource Name (ARN) titik akses dari bucket tujuan sudah benar.
  - Prefiks nama kunci sudah benar. Misalnya, jika Anda mengatur konfigurasi agar mereplikasi objek dengan prefiks `Tax`, maka hanya objek dengan nama kunci seperti `Tax/document1` atau `Tax/document2` yang direplikasi. Objek dengan nama kunci `document3` tidak direplikasi.
  - Statusnya adalah `Enabled`.
- Verifikasi bahwa Penentuan Versi belum ditangguhkan di kedua bucket. Bucket sumber dan tujuan harus memiliki penentuan versi yang diaktifkan.
- Jika bucket tujuan dimiliki oleh yang lain Akun AWS, verifikasi bahwa pemilik bucket memiliki kebijakan bucket pada bucket tujuan yang memungkinkan pemilik bucket sumber untuk mereplikasi

objek. Sebagai contoh, lihat Memberikan izin saat bucket Outposts sumber dan tujuan dimiliki oleh yang berbeda Akun AWS.

- Jika replika objek tidak muncul di bucket tujuan, masalah berikut mungkin mencegah replikasi:
  - S3 on Outposts tidak mereplikasi objek di bucket sumber yang merupakan replika yang dibuat oleh konfigurasi replikasi lain. Misalnya, jika Anda mengatur konfigurasi replikasi dari bucket A ke bucket B ke bucket C, S3 on Outposts tidak mereplikasi replika objek di bucket B ke bucket C.

    Jika Anda ingin mereplikasi objek di bucket A ke bucket B dan bucket C, tetapkan beberapa tujuan bucket dalam aturan replikasi berbeda untuk konfigurasi replikasi bucket sumber Anda. Misalnya, buat dua aturan replikasi pada bucket sumber A, dengan satu aturan untuk direplikasi ke bucket tujuan B dan aturan lainnya untuk direplikasi ke bucket tujuan C.
  - Pemilik bucket sumber dapat memberikan Akun AWS izin lain untuk mengunggah objek. Secara default, pemilik bucket sumber tidak memiliki izin untuk objek yang dibuat oleh akun lain. Konfigurasi replikasi hanya mereplikasi objek yang izin aksesnya dimiliki pemilik bucket sumber. Untuk menghindari masalah replikasi, pemilik bucket sumber dapat memberikan Akun AWS izin lain untuk membuat objek secara kondisional, yang memerlukan izin akses eksplisit pada objek tersebut.
- Misalkan bahwa dalam konfigurasi replikasi, Anda menambahkan aturan untuk mereplikasi subset objek yang memiliki tag tertentu. Dalam kasus ini, Anda harus menetapkan kunci dan nilai tag spesifik pada saat objek dibuat agar S3 on Outposts mereplikasi objek. Jika Anda membuat objek terlebih dahulu lalu menambahkan tag ke objek yang sudah ada, S3 on Putposts tidak akan mereplikasi objek.
- Replikasi gagal jika kebijakan bucket menghalangi akses ke peran replikasi untuk tindakan-tindakan berikut:

  Bucket sumber:

```
"s3-outposts:GetObjectVersionForReplication",
"s3-outposts:GetObjectVersionTagging"
```

  Bucket tujuan:

```
"s3-outposts:ReplicateObject",
"s3-outposts:ReplicateDelete",
"s3-outposts:ReplicateTags"
```

- Amazon EventBridge dapat memberi tahu Anda ketika objek tidak mereplikasi ke Outposts tujuan mereka. Untuk informasi selengkapnya, lihat Menggunakan EventBridge untuk Replikasi S3 di Outposts.

Menggunakan EventBridge untuk Replikasi S3 di Outposts

Amazon S3 di Outposts terintegrasi dengan Amazon EventBridge dan menggunakan namespace. `s3-outposts` EventBridge adalah layanan bus acara tanpa server yang dapat Anda gunakan untuk menghubungkan aplikasi Anda dengan data dari berbagai sumber. Untuk informasi lebih lanjut, lihat Apa itu Amazon EventBridge? di Panduan EventBridge Pengguna Amazon.

Untuk membantu memecahkan masalah konfigurasi replikasi, Anda dapat mengatur Amazon EventBridge untuk menerima pemberitahuan tentang peristiwa kegagalan replikasi. EventBridge dapat memberi tahu Anda dalam kasus ketika objek tidak mereplikasi ke Outposts tujuan mereka. Untuk informasi selengkapnya tentang status saat ini objek yang sedang direplikasi, lihat Gambaran status replikasi.

Setiap kali peristiwa tertentu terjadi di ember Outposts Anda, S3 di Outposts dapat mengirim acara ke. EventBridge Tidak seperti tujuan lainnya, Anda tidak perlu memilih jenis peristiwa yang ingin Anda kirimkan. Anda juga dapat menggunakan EventBridge aturan untuk merutekan acara ke target tambahan. Setelah EventBridge diaktifkan, S3 di Outposts mengirimkan semua peristiwa berikut ke. EventBridge

Jenis peristiwa	Deskripsi	Namespace
`Operation FailedReplication`	Replikasi objek dalam aturan replikasi gagal. Untuk informasi selengkapnya tentang alasan kegagalan S3 di Outposts, lihat Menggunakan EventBridge untuk melihat Replikasi S3 pada alasan kegagalan Outposts.	`s3-outposts`

Menggunakan EventBridge untuk melihat Replikasi S3 pada alasan kegagalan Outposts

Tabel berikut mencantumkan penyebab kegagalan Replikasi S3 pada Outpost. Anda dapat mengonfigurasi EventBridge aturan untuk mempublikasikan dan melihat alasan kegagalan melalui Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS),

atau Amazon Logs. AWS Lambda CloudWatch Untuk informasi selengkapnya tentang izin yang diperlukan untuk menggunakan sumber daya ini EventBridge, lihat Menggunakan kebijakan berbasis sumber daya untuk. EventBridge

Penyebab kegagalan replikasi	Deskripsi
`AssumeRoleNotPermitted`	S3 di Outposts tidak dapat mengasumsikan AWS Identity and Access Management peran (IAM) yang ditentukan dalam konfigurasi replikasi.
`DstBucketNotFound`	S3 di Outposts tidak dapat menemukan bucket tujuan yang ditentukan dalam konfigurasi replikasi.
`DstBucketUnversioned`	Penentuan Versi tidak diaktifkan pada bucket tujuan Outposts. Untuk mereplikasi objek dengan Replikasi S3 di Outposts, Anda harus mengaktifkan Penentuan Versi pada bucket tujuan.
`DstDelObjNotPermitted`	S3 di Outposts tidak dapat mereplikasi penghapusan ke bucket tujuan. Izin `s3-outposts:ReplicateDelete` mungkin tidak ada untuk bucket tujuan.
`DstMultipartCompleteNotPermitted`	S3 pada Outpost tidak dapat menyelesaikan pengunggahan multibagian dari objek di bucket tujuan. Izin `s3-outposts:ReplicateObject` mungkin tidak ada untuk bucket tujuan.
`DstMultipartInitNotPermitted`	S3 di Outposts tidak dapat memulai pengunggahan multibagian objek ke bucket tujuan. Izin `s3-outposts:ReplicateObject` mungkin tidak ada untuk bucket tujuan.

Penyebab kegagalan replikasi	Deskripsi
`DstMultipartPartUploadNotPermitted`	S3 di Outposts tidak dapat mengunggah objek unggahan multipart di bucket tujuan. Izin `s3-outposts:ReplicateObject` mungkin tidak ada untuk bucket tujuan.
`DstOutOfCapacity`	S3 pada Outpost tidak dapat mereplikasi ke Outpost tujuan karena Outpost tersebut kehabisan kapasitas penyimpanan S3.
`DstPutObjNotPermitted`	S3 di Outposts tidak dapat mereplikasi objek ke bucket tujuan. Izin `s3-outposts:ReplicateObject` mungkin tidak ada untuk bucket tujuan.
`DstPutTaggingNotPermitted`	S3 di Outposts tidak dapat mereplikasi tanda objek ke bucket tujuan. Izin `s3-outposts:ReplicateObject` mungkin tidak ada untuk bucket tujuan.
`DstVersionNotFound`	S3 di Outposts tidak dapat menemukan versi objek yang diperlukan di bucket tujuan untuk mereplikasi metadata versi objek tersebut.
`SrcBucketReplicationConfigMissing`	S3 di Outposts tidak dapat menemukan konfigurasi replikasi untuk titik akses yang terkait dengan bucket Outposts sumber.
`SrcGetObjNotPermitted`	S3 di Outposts tidak dapat mengakses objek di bucket sumber untuk replikasi. Izin `s3-outposts:GetObjectVersionForReplication` mungkin tidak ada untuk bucket sumber.

Penyebab kegagalan replikasi	Deskripsi
`SrcGetTaggingNotPermitted`	S3 di Outposts tidak dapat mengakses informasi tag objek dari bucket sumber. Izin `s3-outposts:GetObjectVersionTagging` mungkin tidak ada untuk bucket sumber.
`SrcHeadObjectNotPermitted`	S3 di Outposts tidak dapat mengambil metadata objek dari bucket sumber. Izin `s3-outposts:GetObjectVersionForReplication` mungkin tidak ada untuk bucket sumber.
`SrcObjectNotEligible`	Objek tidak memenuhi syarat untuk replikasi. Objek atau tag objeknya tidak cocok dengan konfigurasi replikasi.

Untuk informasi selengkapnya tentang replikasi pemecahan masalah, lihat topik berikut:

- Membuat peran IAM
- Memecahkan masalah replikasi

Pemantauan EventBridge dengan CloudWatch

Untuk pemantauan, Amazon EventBridge terintegrasi dengan Amazon CloudWatch. EventBridge secara otomatis mengirimkan metrik ke CloudWatch setiap menit. Metrik ini mencakup jumlah peristiwa yang telah dicocokkan dengan aturan dan berapa kali target dipanggil oleh aturan. Ketika aturan berjalan EventBridge, semua target yang terkait dengan aturan dipanggil. Anda dapat memantau EventBridge perilaku Anda melalui CloudWatch cara-cara berikut.

- Anda dapat memantau EventBridgemetrik yang tersedia untuk EventBridge aturan Anda dari CloudWatch dasbor. Kemudian, Anda dapat menggunakan CloudWatch fitur, seperti CloudWatch alarm, untuk mengatur alarm pada metrik tertentu. Jika metrik tersebut mencapai nilai ambang batas khusus yang telah ditentukan dalam alarm, Anda akan menerima pemberitahuan dan dapat melakukan tindakan yang sesuai.

- Anda dapat menetapkan CloudWatch Log Amazon sebagai target EventBridge aturan Anda. Kemudian, EventBridge buat aliran log dan CloudWatch Log menyimpan teks dari peristiwa sebagai entri log. Untuk informasi selengkapnya, lihat EventBridge dan CloudWatch Log.

Untuk informasi selengkapnya tentang EventBridge acara pengiriman dan pengarsipan acara debugging, lihat topik berikut:

- Kebijakan percobaan ulang peristiwa dan menggunakan antrean surat mati
- Acara pengarsipan EventBridge

Berbagi S3 di Outposts dengan menggunakan AWS RAM

Amazon S3 di Outposts mendukung berbagi kapasitas S3 di beberapa akun dalam organisasi dengan menggunakan (). AWS Resource Access Manager AWS RAM Dengan berbagi S3 on Outposts, Anda dapat mengizinkan orang lain membuat dan mengelola bucket, titik akhir, dan titik akses di Outpost Anda.

Topik ini menunjukkan cara menggunakan AWS RAM untuk berbagi S3 di Outposts dan sumber daya terkait dengan yang lain Akun AWS di organisasi Anda. AWS

Prasyarat

- Akun pemilik Outpost memiliki organisasi yang dikonfigurasi. AWS Organizations Untuk informasi selengkapnya, lihat Membuat organisasi di Panduan AWS Organizations Pengguna.
- Organisasi ini menyertakan Akun AWS yang ingin Anda bagikan kapasitas S3 on Outposts Anda. Untuk informasi selengkapnya, lihat Mengirim undangan ke Akun AWS dalam AWS Organizations Panduan Pengguna.
- Pilih salah satu opsi berikut yang ingin Anda bagikan. Sumber daya kedua (baik Subnet atau Outposts) harus dipilih sehingga titik akhir juga dapat diakses. Titik akhir adalah persyaratan jaringan untuk mengakses data yang disimpan di S3 di Outposts.

Opsi 1	Opsi 2
S3 di Outposts	S3 di Outposts

Opsi 1	Opsi 2
Memungkinkan pengguna membuat bucket di Outposts dan titik akses Anda dan menambahkan objek ke bucket tersebut.	Memungkinkan pengguna membuat bucket di Outposts dan titik akses Anda dan menambahkan objek ke bucket tersebut.
Subnet	Outposts
Memungkinkan pengguna untuk menggunak an virtual private cloud (VPC) dan titik akhir yang terkait dengan subnet Anda.	Memungkinkan pengguna untuk melihat bagan kapasitas S3 dan halaman beranda AWS Outposts konsol. Juga memungkinkan pengguna untuk membuat subnet di Outposts bersama dan membuat titik akhir.

Prosedur

1. Masuk ke AWS Management Console dengan menggunakan Akun AWS yang memiliki Outpost, dan kemudian buka AWS RAM konsol di https://console.aws.amazon.com/ram/ rumah.
2. Pastikan Anda telah mengaktifkan berbagi dengan AWS Organizations in AWS RAM. Untuk selengkapnya, lihat Mengaktifkan berbagi sumber daya AWS Organizations di dalam Panduan AWS RAM Pengguna.
3. Gunakan Opsi 1 atau Opsi 2 dalam prasyarat untuk membuat pembagian sumber daya. Jika Anda memiliki beberapa sumber daya S3 di Outposts, pilih Amazon Resource Names ARNs () dari sumber daya yang ingin Anda bagikan. Untuk mengaktifkan titik akhir, bagikan subnet atau Outpost Anda.

   Untuk informasi selengkapnya tentang cara membuat pembagian sumber daya, lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.
4. Akun AWS Yang Anda bagikan sumber daya Anda sekarang harus dapat menggunakan S3 di Outposts. Bergantung pada opsi yang Anda pilih dalam prasyarat, berikan informasi berikut kepada pengguna akun:

Opsi 1	Opsi 2
Outpost ID	Outpost ID

Opsi 1	Opsi 2
ID VPC ID subnet. ID dari grup keamanan	

> **Note**
>
> Pengguna dapat mengonfirmasi bahwa sumber daya telah dibagikan dengan mereka menggunakan AWS RAM konsol, AWS Command Line Interface (AWS CLI) AWS SDKs, atau REST API. Pengguna dapat melihat pembagian sumber daya yang ada dengan menggunakan perintah get-resource-sharesCLI.

Contoh penggunaan

Setelah Anda membagikan sumber daya S3 di Outposts dengan akun lain, akun tersebut dapat mengelola bucket dan objek di Outpost Anda. Jika Anda membagikan sumber daya Subnet, maka akun tersebut dapat menggunakan titik akhir yang Anda buat. Contoh berikut menunjukkan bagaimana pengguna dapat menggunakan AWS CLI untuk berinteraksi dengan Outpost Anda setelah Anda membagikan sumber daya ini.

Example Buat bucket

Contoh berikut membuat bucket bernama *amzn-s3-demo-bucket1* di Outpost*op-01ac5d28a6a232904*. Sebelum menggunakan perintah ini, ganti masing-masing *user input placeholder* dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3control create-bucket --bucket amzn-s3-demo-bucket1 --outpost-
id op-01ac5d28a6a232904
```

Untuk informasi selengkapnya tentang perintah ini, lihat create-bucket di Referensi.AWS CLI

Example Membuat Titik Akses

Contoh berikut membuat titik akses pada Outpost dengan menggunakan parameter contoh dalam tabel berikut. Sebelum menggunakan perintah ini, ganti *user input placeholder* nilai-nilai ini dan Wilayah AWS kode dengan nilai yang sesuai untuk kasus penggunaan Anda.

Parameter	Nilai
account-id	*111122223333*
Nama titik akses	*example-outpost-access-point*
outpost-id	*op-01ac5d28a6a232904*
Nama bucket outpost	*amzn-s3-demo-bucket1*
ID VPC	*vpc-1a2b3c4d5e6f7g8h9*

> **Note**
>
> Parameter ID Akun harus berupa Akun AWS ID pemilik bucket, yang merupakan pengguna bersama.

```
aws s3control create-access-point --account-id 111122223333 --name example-outpost-
access-point \
--bucket arn:aws:s3-outposts:us-east-1:111122223333:outpost/op-01ac5d28a6a232904/
bucket/amzn-s3-demo-bucket1 \
--vpc-configuration VpcId=vpc-1a2b3c4d5e6f7g8h9
```

Untuk informasi selengkapnya tentang perintah ini, lihat create-access-pointdi AWS CLI Referensi.

Example Mengunggah objek

Contoh berikut mengunggah file *my_image.jpg* dari sistem file lokal pengguna ke objek bernama *images/my_image.jpg* melalui titik akses *example-outpost-access-point* di Outpost *op-01ac5d28a6a232904*, yang dimiliki oleh akun AWS *111122223333*. Sebelum menggunakan perintah ini, ganti *user input placeholder* nilai-nilai ini dan Wilayah AWS kode dengan nilai yang sesuai untuk kasus penggunaan Anda.

```
aws s3api put-object --bucket arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/example-outpost-access-
point \
--body my_image.jpg --key images/my_image.jpg
```

Untuk informasi selengkapnya tentang perintah ini, lihat put-object dalam Referensi AWS CLI .

> **Note**
>
> Jika operasi ini menghasilkan kesalahan Sumber Daya tidak ditemukan atau tidak responsif, VPC Anda mungkin tidak memiliki titik akhir bersama.
> Untuk memeriksa apakah ada titik akhir bersama, gunakan list-shared-endpoints AWS CLI perintah. Jika tidak ada titik akhir bersama, bekerja dengan pemilik Outpost untuk membuatnya. Untuk informasi selengkapnya, lihat ListSharedEndpoints dalam Referensi API Amazon Simple Storage Service.

Example : Membuat titik akhir

Berikut adalah contoh membuat titik akhir di Outpost bersama. Sebelum menggunakan perintah ini, ganti *`user input placeholder`* nilai untuk ID Outpost, ID subnet, dan ID grup keamanan dengan nilai yang sesuai untuk kasus penggunaan Anda.

> **Note**
>
> Pengguna dapat melakukan operasi ini hanya jika pembagian sumber daya menyertakan sumber daya Outposts.

```
aws s3outposts create-endpoint --outposts-id op-01ac5d28a6a232904 --subnet-id XXXXXX --
security-group-id XXXXXXX
```

Untuk informasi selengkapnya tentang perintah ini, lihat buat-titik akhir di Referensi.AWS CLI

Lainnya Layanan AWS yang menggunakan S3 di Outposts

Lain Layanan AWS yang berjalan lokal untuk Anda juga AWS Outposts dapat menggunakan Amazon S3 Anda pada kapasitas Outposts. Di Amazon CloudWatch , `S3Outposts` namespace menampilkan

metrik terperinci untuk bucket dalam S3 di Outposts, tetapi metrik ini tidak menyertakan penggunaan untuk yang lain. Layanan AWS Untuk mengelola kapasitas S3 Anda pada Outposts yang dikonsumsi oleh orang Layanan AWS lain, lihat informasi dalam tabel berikut.

Layanan AWS	Deskripsi	Pelajari selengkapnya
Amazon S3	Semua penggunaan S3 langsung pada Outposts memiliki akun yang cocok dan CloudWatch metrik bucket.	Lihat metrik
Amazon Elastic Block Store (Amazon EBS)	Untuk Amazon EBS di Outposts, Anda dapat memilih AWS Outpost sebagai tujuan snapshot Anda dan menyimpan secara lokal di S3 Anda di Outpost.	Pelajari selengkapnya
Amazon Relationa l Database Service (Amazon RDS)	Anda dapat menggunakan cadangan lokal Amazon RDS untuk menyimpan cadangan RDS Anda secara lokal di Outpost Anda.	Pelajari selengkapnya

Memantau S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, silakan lihat Apa itu Amazon S3 di Outposts?

Untuk informasi selengkapnya tentang memantau Amazon S3 di kapasitas penyimpanan Outposts, lihat topik-topik berikut.

Topik

- Mengelola kapasitas S3 pada Outposts dengan metrik Amazon CloudWatch
- Menerima pemberitahuan acara S3 di Outposts dengan menggunakan Amazon Events CloudWatch
- Memantau S3 di AWS CloudTrail Outposts dengan log

Mengelola kapasitas S3 pada Outposts dengan metrik Amazon CloudWatch

Untuk membantu mengelola kapasitas S3 tetap di Outpost Anda, kami sarankan Anda membuat CloudWatch peringatan yang memberi tahu Anda kapan pemanfaatan penyimpanan Anda melebihi ambang batas tertentu. Untuk informasi selengkapnya tentang CloudWatch metrik untuk S3 di Outposts, lihat. CloudWatch metrik Jika tidak tersedia cukup ruang untuk menyimpan objek di Outpost Anda, API mengembalikan pengecualian kapasitas yang tidak memadai (ICE). Untuk mengosongkan ruang, Anda dapat membuat CloudWatch alarm yang memicu penghapusan data eksplisit, atau menggunakan kebijakan kedaluwarsa siklus hidup untuk objek kedaluwarsa. Untuk menyimpan data sebelum dihapus, Anda dapat menggunakannya AWS DataSync untuk menyalin data dari bucket Amazon S3 di Outposts ke bucket S3 di file. Wilayah AWS Untuk informasi

selengkapnya tentang penggunaan DataSync, lihat Memulai AWS DataSync di Panduan AWS DataSync Pengguna.

CloudWatch metrik

Namespace `S3Outposts` mencakup metrik berikut untuk Amazon S3 pada bucket Outposts. Anda dapat memantau jumlah total S3 di byte Outposts yang disediakan, total byte gratis yang tersedia untuk objek, dan ukuran total semua objek untuk bucket tertentu. Metrik terkait bucket atau akun ada untuk semua penggunaan S3 langsung. Penggunaan S3 tidak langsung, seperti menyimpan snapshot lokal Amazon Elastic Block Store atau backup Amazon Relational Database Service di Outpost, menghabiskan kapasitas S3, tetapi tidak disertakan dalam bucket atau metrik terkait akun. Untuk informasi selengkapnya tentang snapshot lokal Amazon EBS, lihat snapshot lokal Amazon EBS di Outposts. Untuk melihat laporan biaya Amazon EBS Anda, kunjungi https://console.aws.amazon.com/costmanagement/.

> **Note**
>
> S3 di Outposts hanya men support metrik berikut dan bukan metrik Amazon S3 lainnya. Karena S3 di Outposts memiliki batas kapasitas tetap, kami sarankan CloudWatch membuat alarm untuk memberi tahu Anda ketika penggunaan penyimpanan Anda melebihi ambang batas tertentu.

Metrik	Deskripsi	Periode Waktu	Unit	Jenis
`Outpost` `talByte`	Total kapasitas yang disediaka n dalam byte untuk Outposts.	5 menit	Byte	S3 di Outposts
`Outpost` `eeBytes`	Jumlah byte gratis yang tersedia di Outposts untuk menyimpan data pelanggan.	5 menit	Byte	S3 di Outposts
`BucketU` `dBytes`	Ukuran total semua objek untuk bucket yang diberikan.	5 menit	Byte	S3 di Outposts. Penggunaan S3 langsung saja.

Metrik	Deskripsi	Periode Waktu	Unit	Jenis
AccountUsedBytes	Ukuran total semua objek untuk akun Outposts yang ditentukan.	5 menit	Byte	S3 di Outposts. Penggunaan S3 langsung saja.
BytesPendingReplication	Jumlah total byte objek yang menunggu replikasi untuk aturan replikasi yang diberikan. Untuk informasi selengkapnya tentang cara mengaktifkan metrik replikasi, lihat Membuat aturan replikasi di antara Outposts.	5 menit	Byte	Tidak wajib. Untuk Replikasi S3 di Outposts.
OperationsPendingReplication	Jumlah total operasi yang menunggu replikasi untuk aturan replikasi yang diberikan. Untuk informasi selengkapnya tentang cara mengaktifkan metrik replikasi, lihat Membuat aturan replikasi di antara Outposts.	5 menit	Hitungan	Tidak wajib. Untuk Replikasi S3 di Outposts.
ReplicationLatency	Jumlah penundaan saat ini saat bucket tujuan replikasi berada di belakang bucket sumber untuk aturan replikasi yang diberikan. Untuk informasi selengkapnya tentang cara mengaktifkan metrik replikasi, lihat Membuat aturan replikasi di antara Outposts.	5 menit	Detik	Tidak wajib. Untuk Replikasi S3 di Outposts.

Menerima pemberitahuan acara S3 di Outposts dengan menggunakan Amazon Events CloudWatch

Anda dapat menggunakan CloudWatch Acara untuk membuat aturan untuk Amazon S3 pada peristiwa API Outposts. Saat membuat aturan, Anda dapat memilih untuk mendapatkan pemberitahuan melalui semua CloudWatch target yang didukung, termasuk Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS), dan. AWS Lambda Untuk informasi selengkapnya, lihat daftar AWS layanan yang dapat menjadi target untuk CloudWatch Acara di Panduan Pengguna CloudWatch Acara Amazon. Untuk memilih layanan target agar berfungsi dengan S3 Anda di Outposts, lihat Membuat aturan Peristiwa CloudWatch yang memicu panggilan AWS CloudTrail API AWS menggunakan Panduan Pengguna CloudWatch Amazon Events.

> **Note**
>
> Untuk S3 pada operasi objek Outposts AWS , peristiwa panggilan API yang dikirim CloudTrail oleh akan cocok dengan aturan Anda hanya jika Anda memiliki jejak (opsional dengan pemilih peristiwa) yang dikonfigurasi untuk menerima peristiwa tersebut. Untuk informasi selengkapnya, lihat Bekerja dengan file CloudTrail log di Panduan AWS CloudTrail Pengguna.

Example

Berikut ini adalah contoh peraturan untuk operasi `DeleteObject`. Untuk menggunakan aturan contoh ini, ganti *`amzn-s3-demo-bucket1`* dengan nama S3 Anda di bucket Outposts.

```
{
  "source": [
    "aws.s3-outposts"
  ],
  "detail-type": [
    "AWS API call through CloudTrail"
  ],
  "detail": {
    "eventSource": [
      "s3-outposts.amazonaws.com"
    ],
    "eventName": [
```

```
      "DeleteObject"
    ],
    "requestParameters": {
      "bucketName": [
        "amzn-s3-demo-bucket1"
      ]
    }
  }
}
```

Memantau S3 di AWS CloudTrail Outposts dengan log

Amazon S3 di Outposts terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau di Layanan AWS S3 di Outposts. Anda dapat menggunakan AWS CloudTrail untuk mendapatkan informasi tentang S3 tentang permintaan tingkat bucket Outposts dan tingkat objek untuk mengaudit dan mencatat aktivitas acara S3 Anda di Outposts.

Untuk mengaktifkan peristiwa CloudTrail data untuk semua bucket Outposts Anda atau untuk daftar bucket Outposts tertentu, Anda harus membuat jejak secara manual. CloudTrail Untuk informasi selengkapnya tentang entri file CloudTrail log, lihat S3 pada entri file log Outposts.

Untuk daftar lengkap peristiwa CloudTrail data untuk S3 di Outposts, lihat peristiwa data Amazon S3 di Panduan Pengguna Amazon CloudTrail S3.

> **Note**
>
> - Merupakan praktik terbaik untuk membuat kebijakan siklus hidup untuk bucket Outposts peristiwa AWS CloudTrail data Anda. Konfigurasikan kebijakan siklus aktif untuk menghapus berkas log secara berkala setelah periode saat Anda perlu mengauditnya. Hal ini mengurangi jumlah data yang dianalisis Amazon Athena untuk setiap kueri. Untuk informasi selengkapnya, lihat Membuat dan mengelola konfigurasi siklus hidup untuk bucket Amazon S3 di Outposts Anda.
> - Untuk contoh cara menanyakan CloudTrail log, lihat posting Blog AWS Big Data Menganalisis Keamanan, Kepatuhan, dan Penggunaan Aktivitas Operasional AWS CloudTrail dan Amazon Athena.

Aktifkan CloudTrail logging untuk objek di bucket S3 pada Outposts

Anda dapat menggunakan konsol Amazon S3 untuk mengonfigurasi AWS CloudTrail jejak untuk mencatat peristiwa data untuk objek di bucket Amazon S3 di Outposts. CloudTrail mendukung logging S3 pada operasi API tingkat objek Outposts seperti`GetObject`,, dan. `DeleteObject` `PutObject` Peristiwa ini disebut peristiwa data.

Secara default, CloudTrail jejak tidak mencatat peristiwa data. Namun, Anda dapat mengonfigurasi jejak untuk mencatat peristiwa data untuk S3 pada bucket Outposts yang Anda tentukan atau untuk mencatat peristiwa data untuk semua bucket S3 di Outposts di Anda. Akun AWS

CloudTrail tidak mengisi peristiwa data dalam riwayat CloudTrail acara. Selain itu, tidak semua operasi API tingkat ember S3 di Outposts diisi dalam riwayat acara. CloudTrail Untuk informasi selengkapnya tentang cara menanyakan CloudTrail log, lihat Menggunakan pola filter CloudWatch Log Amazon dan Amazon Athena untuk menanyakan CloudTrail log di Pusat AWS Pengetahuan.

Untuk mengonfigurasi jejak guna mencatat peristiwa data bagi bucket S3 di Outposts, Anda dapat menggunakan konsol AWS CloudTrail tersebut atau konsol Amazon S3. Jika Anda mengonfigurasi jejak untuk mencatat peristiwa data untuk semua bucket S3 on Outposts di Anda Akun AWS, lebih mudah menggunakan konsol. CloudTrail Untuk informasi tentang menggunakan CloudTrail konsol untuk mengonfigurasi jejak untuk mencatat peristiwa data S3 di Outposts, lihat Peristiwa data di Panduan Pengguna AWS CloudTrail .

> **Important**
>
> Biaya tambahan berlaku untuk peristiwa data. Untuk informasi selengkapnya, lihat harga AWS CloudTrail.

Prosedur berikut menunjukkan cara menggunakan konsol Amazon S3 untuk mengonfigurasi CloudTrail jejak untuk mencatat peristiwa data untuk bucket S3 di Outposts.

> **Note**
>
> Akun AWS Yang membuat bucket memilikinya dan merupakan satu-satunya yang dapat mengonfigurasi peristiwa data S3 pada Outposts yang akan dikirim. AWS CloudTrail

Untuk mengaktifkan pencatatan peristiwa CloudTrail data untuk objek di bucket S3 di Outposts

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Di panel navigasi kiri, pilih bucket Outposts.
3. Pilih nama bucket Outposts yang peristiwa datanya ingin Anda log dengan menggunakan. CloudTrail
4. Pilih Properti.
5. Di bagian peristiwa AWS CloudTrail data, pilih Configure in CloudTrail.

   AWS CloudTrail Konsol terbuka.

   Anda dapat membuat CloudTrail jejak baru atau menggunakan kembali jejak yang ada dan mengonfigurasi peristiwa data S3 pada Outposts untuk dicatat di jejak Anda.
6. Pada halaman Dasbor CloudTrail konsol, pilih Buat jejak.
7. Pada halaman Langkah 1 Pilih atribut jejak, berikan nama untuk jejak, pilih bucket S3 untuk menyimpan log jejak, tentukan pengaturan lain yang Anda inginkan, lalu pilih Berikutnya.
8. Pada Langkah 2 Pilih halaman peristiwa log, di bawah Jenis acara, pilih Peristiwa data.

   Untuk jenis peristiwa Data, pilih Outposts S3. Pilih Selanjutnya.

> **Note**
>
> - Saat Anda membuat jejak dan mengonfigurasi pencatatan peristiwa data untuk S3 di Outposts, Anda harus menentukan jenis peristiwa data dengan benar.
>   - Jika Anda menggunakan CloudTrail konsol, pilih Outposts S3 untuk jenis peristiwa Data. Untuk informasi tentang cara membuat jejak di CloudTrail konsol, lihat Membuat dan memperbarui jejak dengan konsol di Panduan AWS CloudTrail Pengguna. Untuk informasi tentang cara mengonfigurasi log peristiwa data S3 di Outposts di CloudTrail konsol, lihat Mencatat peristiwa data untuk Objek Amazon S3 di Panduan Pengguna.AWS CloudTrail
>   - Jika Anda menggunakan AWS Command Line Interface (AWS CLI) atau AWS SDKs, atur `resources.type` bidang ke`AWS::S3Outposts::Object`. Untuk informasi selengkapnya tentang cara mencatat peristiwa data S3 di Outposts dengan AWS CLI, lihat Log S3 pada peristiwa Outposts di Panduan Pengguna.AWS CloudTrail

- Jika Anda menggunakan CloudTrail konsol atau konsol Amazon S3 untuk mengonfigurasi jejak untuk mencatat peristiwa data untuk bucket S3 di Outposts, konsol Amazon S3 menunjukkan bahwa pencatatan tingkat objek diaktifkan untuk bucket.

9. Pada halaman Langkah 3 Tinjau dan buat, tinjau atribut jejak dan peristiwa log yang Anda konfigurasikan. Kemudian, pilih Buat jejak.

Untuk menonaktifkan pencatatan peristiwa CloudTrail data untuk objek di bucket S3 di Outposts

1. Masuk ke AWS Management Console dan buka CloudTrail konsol di https://console.aws.amazon.com/cloudtrail/.
2. Di panel navigasi kiri, pilih Jalur.
3. Pilih nama jejak yang Anda buat untuk mencatat peristiwa untuk bucket S3 on Outposts Anda.
4. Di halaman detail untuk jejak Anda, pilih Hentikan pencatatan di sudut kanan atas.
5. Pada kotak dialog yang muncul, pilih Hentikan pencatatan.

Amazon S3 pada entri file log AWS CloudTrail Outposts

Amazon S3 pada acara manajemen Outposts tersedia melalui. AWS CloudTrail Selain itu, Anda dapat secara opsional mengaktifkan pencatatan untuk peristiwa data di AWS CloudTrail.

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket S3 di Wilayah yang Anda tentukan. CloudTrail log untuk bucket Outposts Anda menyertakan bidang baru`edgeDeviceDetails`, yang mengidentifikasi Outpost tempat bucket yang ditentukan berada.

Bidang log tambahan mencakup tindakan yang diminta, tanggal dan waktu tindakan, dan parameter permintaan. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan PutObjecttindakan pada`s3-outposts`.

```
{
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
```

```
        "arn": "arn:aws:iam::111122223333:user/yourUserName",
        "accountId": "222222222222",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "yourUserName"
      },
      "eventTime": "2020-11-30T15:44:33Z",
      "eventSource": "s3-outposts.amazonaws.com",
      "eventName": "PutObject",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "26.29.66.20",
      "userAgent": "aws-cli/1.18.39 Python/3.4.10 Darwin/18.7.0 botocore/1.15.39",
      "requestParameters": {
        "expires": "Wed, 21 Oct 2020 07:28:00 GMT",
        "Content-Language": "english",
        "x-amz-server-side-encryption-customer-key-MD5": "wJalrXUtnFEMI/K7MDENG/
bPxRfiCYEXAMPLEKEY",
        "ObjectCannedACL": "BucketOwnerFullControl",
        "x-amz-server-side-encryption": "Aes256",
        "Content-Encoding": "gzip",
        "Content-Length": "10",
        "Cache-Control": "no-cache",
        "Content-Type": "text/html; charset=UTF-8",
        "Content-Disposition": "attachment",
        "Content-MD5": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
        "x-amz-storage-class": "Outposts",
        "x-amz-server-side-encryption-customer-algorithm": "Aes256",
        "bucketName": "amzn-s3-demo-bucket1",
        "Key": "path/upload.sh"
      },
      "responseElements": {
        "x-amz-server-side-encryption-customer-key-MD5": "wJalrXUtnFEMI/K7MDENG/
bPxRfiCYEXAMPLEKEY",
        "x-amz-server-side-encryption": "Aes256",
        "x-amz-version-id": "001",
        "x-amz-server-side-encryption-customer-algorithm": "Aes256",
        "ETag": "d41d8cd98f00b204e9800998ecf8427f"
      },
      "additionalEventData": {
        "CipherSuite": "ECDHE-RSA-AES128-SHA",
        "bytesTransferredIn": 10,
        "x-amz-id-2": "29xXQBV2O
+xOHKItvzY1suLv1i6A52E0zOX159fpfsItYd58JhXwKxXAXI4IQkp6",
        "SignatureVersion": "SigV4",
        "bytesTransferredOut": 20,
```

```
        "AuthenticationMethod": "AuthHeader"
      },
      "requestID": "8E96D972160306FA",
      "eventID": "ee3b4e0c-ab12-459b-9998-0a5a6f2e4015",
      "readOnly": false,
      "resources": [
        {
          "accountId": "222222222222",
          "type": "AWS::S3Outposts::Object",
          "ARN": "arn:aws:s3-outposts:us-east-1:YYY:outpost/op-01ac5d28a6a232904/
bucket/path/upload.sh"
        },
        {
          "accountId": "222222222222",
          "type": "AWS::S3Outposts::Bucket",
          "ARN": "arn:aws:s3-outposts:us-east-1:YYY:outpost/op-01ac5d28a6a232904/
bucket/"
        }
      ],
      "eventType": "AwsApiCall",
      "managementEvent": false,
      "recipientAccountId": "444455556666",
      "sharedEventID": "02759a4c-c040-4758-b84b-7cbaaf17747a",
      "edgeDeviceDetails": {
        "type": "outposts",
        "deviceId": "op-01ac5d28a6a232904"
      },
      "eventCategory": "Data"
    }
```

Mengembangkan dengan Amazon S3 di Outposts

Dengan Amazon S3 di Outposts, Anda dapat membuat bucket S3 di AWS Outposts dan dengan mudah menyimpan dan mengambil objek di tempat untuk aplikasi yang memerlukan akses data lokal, pemrosesan data lokal, dan residensi data. S3 on Outposts menyediakan kelas penyimpanan baru, S3 Outposts 0UTP0STS (), yang menggunakan Amazon APIs S3, dan dirancang untuk menyimpan data secara tahan lama dan berlebihan di beberapa perangkat dan server di perangkat Anda. AWS Outposts Anda berkomunikasi dengan bucket Outposts menggunakan titik akses dan koneksi titik akhir melalui cloud privat virtual (VPC). Anda dapat menggunakan fitur yang sama APIs dan pada bucket Outpost seperti yang Anda lakukan di bucket Amazon S3, termasuk kebijakan akses, enkripsi, dan penandaan. Anda dapat menggunakan S3 di Outposts melalui AWS Management Console AWS Command Line Interface ,AWS CLI() AWS SDKs, atau REST API. Untuk informasi selengkapnya, lihat Apa itu Amazon S3 di Outposts?

Topik berikut menyediakan informasi tentang pengembangan dengan S3 di Outposts.

Topik

S3 di wilayah yang didukung Outposts

S3 di Outposts didukung sebagai berikut. Wilayah AWS

- AS Timur (Virginia Utara) (us-east-1)
- AS Timur (Ohio) (us-east-2)
- AS Barat (California Utara) (us-west-1)
- AS Barat (Oregon) (us-west-2)
- Africa (Cape Town) (af-south-1)
- Asia Pasifik (Jakarta) (ap-sourtheast-3)
- Asia Pasifik (Mumbai) (ap-south-1)
- Asia Pasifik (Osaka) (ap-northeast-3)

- Asia Pasifik (Seoul) (ap-northeast-2)
- Asia Pasifik (Singapura) (ap-southeast-1)
- Asia Pacific (Sydney) (ap-southeast-2)
- Asia Pacific (Tokyo) (ap-northeast-1)
- Kanada (Pusat) (ca-central-1)
- Eropa (Frankfurt) (eu-central-1)
- Eropa (Irlandia) (eu-west-1)
- Eropa (London) (eu-west-2)
- Europe (Milan) (eu-south-1)
- Eropa (Paris) (eu-west-3)
- Eropa (Stockholm) (eu-north-1)
- Israel (Tel Aviv) (il-central-1)
- Middle East (Bahrain) (me-south-1)
- Amerika Selatan (Sao Paulo) (sa-east-1)
- AWS GovCloud (AS-Timur) (us-gov-east-1)
- AWS GovCloud (AS-Barat) (us-gov-west-1)

Operasi API Amazon S3 di Outposts

Topik ini mencantumkan operasi API Amazon S3, Amazon S3 Control, dan Amazon S3 di Outposts yang dapat Anda gunakan dengan Amazon S3 di Outposts.

Topik

Operasi API Amazon S3 untuk mengelola objek

S3 di Outposts dirancang untuk menggunakan objek API yang sama dengan Amazon S3. Anda harus menggunakan titik akses untuk mengakses objek apa pun dalam bucket Outposts. Saat Anda menggunakan operasi API objek dengan S3 di Outposts, Anda memberikan Amazon Resource Name

(ARN) titik akses atau alias titik akses Outposts. Untuk informasi selengkapnya tentang alias titik akses, lihat Menggunakan alias gaya bucket untuk titik akses bucket S3 di Outposts.

Amazon S3 di Outposts mendukung operasi API Amazon S3 berikut:

- AbortMultipartUpload
- CompleteMultipartUpload
- CopyObject
- CreateMultipartUpload
- DeleteObject
- DeleteObjects
- DeleteObjectTagging
- GetObject
- GetObjectTagging
- HeadBucket
- HeadObject
- ListMultipartUploads
- ListObjects
- ListObjectsV2
- ListObjectVersions
- ListParts
- PutObject
- PutObjectTagging
- UploadPart
- UploadPartCopy

Operasi API Amazon S3 Control untuk mengelola bucket

S3 di Outposts mendukung operasi API Amazon S3 Control berikut untuk bekerja dengan bucket.

- CreateAccessPoint
- CreateBucket

- DeleteAccessPoint
- DeleteAccessPointPolicy
- DeleteBucket
- DeleteBucketLifecycleConfiguration
- DeleteBucketPolicy
- DeleteBucketReplication
- DeleteBucketTagging
- GetAccessPoint
- GetAccessPointPolicy
- GetBucket
- GetBucketLifecycleConfiguration
- GetBucketPolicy
- GetBucketReplication
- GetBucketTagging
- GetBucketVersioning
- ListAccessPoints
- ListRegionalBuckets
- PutAccessPointPolicy
- PutBucketLifecycleConfiguration
- PutBucketPolicy
- PutBucketReplication
- PutBucketTagging
- PutBucketVersioning

Operasi API S3 di Outposts untuk mengelola Outposts

S3 di Outposts mendukung operasi Amazon S3 di API Outposts berikut untuk mengelola titik akhir.

- CreateEndpoint
- DeleteEndpoint
- ListEndpoints

- ListOutpostsWithS3
- ListSharedEndpoints

Mengonfigurasi klien kontrol S3 untuk S3 di Outposts menggunakan SDK for Java

Contoh berikut mengonfigurasi klien kontrol Amazon S3 untuk Amazon S3 di Outposts menggunakan AWS SDK untuk Java. Untuk menggunakan contoh ini, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

```
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.services.s3control.AWSS3Control;
import com.amazonaws.services.s3control.AWSS3ControlClient;

public AWSS3Control createS3ControlClient() {

    String accessKey = AWSAccessKey;
    String secretKey = SecretAccessKey;
    BasicAWSCredentials awsCreds = new BasicAWSCredentials(accessKey, secretKey);

    return AWSS3ControlClient.builder().enableUseArnRegion()
            .withCredentials(new AWSStaticCredentialsProvider(awsCreds))
            .build();

}
```

Membuat permintaan ke S3 di Outposts over IPv6

Amazon S3 di Outposts dan S3 di Outposts dual-stack endpoint mendukung permintaan ke S3 pada bucket Outposts menggunakan protokol atau protokol. IPv6 IPv4 Dengan IPv6 dukungan untuk S3 di Outposts, Anda dapat mengakses dan mengoperasikan bucket dan mengontrol sumber daya pesawat melalui S3 on Outposts melalui jaringan. APIs IPv6

> **Note**
>
> Tindakan objek S3 pada Outposts (`PutObject`seperti `GetObject` atau) tidak didukung IPv6 melalui jaringan.

Tidak ada biaya tambahan untuk mengakses S3 di Outposts melalui jaringan. IPv6 Untuk informasi lebih lanjut tentang S3 di Outposts, lihat harga S3 di Outposts.

Topik

Memulai dengan IPv6

Untuk membuat permintaan ke bucket S3 on Outposts, Anda harus menggunakan IPv6 endpoint dual-stack. Bagian selanjutnya menjelaskan cara membuat permintaan IPv6 dengan menggunakan titik akhir dual-stack.

Berikut ini adalah pertimbangan penting sebelum mencoba mengakses bucket S3 on Outposts: IPv6

- Klien dan jaringan yang mengakses bucket harus diaktifkan agar dapat menggunakan IPv6.
- Permintaan gaya host virtual dan gaya jalur didukung untuk akses. IPv6 Untuk informasi selengkapnya, lihat Menggunakan S3 pada titik akhir dual-stack Outposts.
- Jika Anda menggunakan pemfilteran alamat IP sumber di pengguna AWS Identity and Access Management (IAM) atau kebijakan bucket S3 on Outposts, Anda harus memperbarui kebijakan untuk menyertakan rentang alamat. IPv6

  > **Note**
  >
  > Persyaratan ini hanya berlaku untuk operasi bucket S3 pada Outposts dan mengontrol sumber daya IPv6 pesawat di seluruh jaringan. Tindakan objek Amazon S3 di Outposts tidak didukung di seluruh jaringan. IPv6

- Saat menggunakan IPv6, file log akses server mengeluarkan alamat IP dalam IPv6 format. Anda harus memperbarui alat, skrip, dan perangkat lunak yang ada yang Anda gunakan untuk mengurai S3 pada file log Outposts, sehingga mereka dapat mengurai alamat IP jarak jauh yang diformat.

IPv6 Alat, skrip, dan perangkat lunak yang diperbarui kemudian akan mengurai alamat IP jarak jauh yang IPv6 diformat dengan benar.

Menggunakan titik akhir dual-stack untuk membuat permintaan melalui jaringan IPv6

Untuk membuat permintaan dengan S3 pada panggilan Outposts IPv6 API, Anda dapat menggunakan titik akhir dual-stack melalui atau SDK. AWS CLI AWS Operasi API kontrol Amazon S3 dan operasi S3 pada Outposts API bekerja dengan cara yang sama apakah Anda mengakses S3 di Outposts melalui protokol atau protokol. IPv6 IPv4 Namun, ketahuilah bahwa tindakan objek S3 pada Outposts (`PutObject`seperti `GetObject` atau) tidak didukung IPv6 melalui jaringan.

Saat menggunakan AWS Command Line Interface (AWS CLI) dan AWS SDKs, Anda dapat menggunakan parameter atau flag untuk mengubah ke titik akhir dual-stack. Anda juga dapat menentukan titik akhir dual-stack secara langsung sebagai penggantian titik akhir S3 on Outposts dalam file konfigurasi.

Anda dapat menggunakan endpoint dual-stack untuk mengakses bucket S3 di Outposts dari salah satu IPv6 dari berikut ini:

- Itu AWS CLI, lihatMenggunakan titik akhir tumpukan ganda dari AWS CLI.
- Itu AWS SDKs, lihatMenggunakan S3 pada titik akhir tumpukan ganda Outposts dari AWS SDKs.

Menggunakan IPv6 alamat dalam kebijakan IAM

Sebelum mencoba mengakses bucket S3 di Outposts menggunakan IPv6 protokol, pastikan bahwa pengguna IAM atau kebijakan bucket S3 pada Outposts yang digunakan untuk pemfilteran alamat IP diperbarui untuk menyertakan rentang alamat. IPv6 Jika kebijakan pemfilteran alamat IP tidak diperbarui untuk menangani IPv6 alamat, Anda dapat kehilangan akses ke bucket S3 di Outposts saat mencoba menggunakan protokol. IPv6

Kebijakan IAM yang memfilter alamat IP menggunakan operator kondisi alamat IP. Kebijakan bucket S3 on Outposts berikut mengidentifikasi rentang IP 54.240.143.* dari alamat yang diizinkan dengan menggunakan operator kondisi alamat IP. IPv4 Alamat IP apa pun di luar rentang ini akan ditolak aksesnya ke bucket `DOC-EXAMPLE-BUCKET` S3 on Outposts (). Karena semua IPv6 alamat berada di luar rentang yang diizinkan, kebijakan ini mencegah IPv6 alamat agar tidak dapat diakses`DOC-EXAMPLE-BUCKET`.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3-outposts:*",
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/OUTPOSTS-ID/bucket/DOC-EXAMPLE-BUCKET/*",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": "54.240.143.0/24"
                }
            }
        }
    ]
}
```

Anda dapat memodifikasi elemen kebijakan `Condition` bucket S3 pada Outposts untuk mengizinkan IPv4 rentang alamat `54.240.143.0/24` () IPv6 dan `2001:DB8:1234:5678::/64` () seperti yang ditunjukkan pada contoh berikut. Anda dapat menggunakan tipe blok `Condition` yang ditampilkan dalam contoh untuk memperbarui kebijakan pengguna IAM dan bucket Anda.

```
"Condition": {
  "IpAddress": {
     "aws:SourceIp": [
       "54.240.143.0/24",
        "2001:DB8:1234:5678::/64"
      ]
   }
 }
```

Sebelum menggunakan, IPv6 Anda harus memperbarui semua kebijakan pengguna dan bucket IAM yang relevan yang menggunakan pemfilteran alamat IP untuk mengizinkan rentang IPv6 alamat. Kami menyarankan Anda memperbarui kebijakan IAM dengan rentang IPv6 alamat organisasi Anda selain rentang IPv4 alamat yang ada. Untuk contoh kebijakan bucket yang mengizinkan akses melalui keduanya IPv6 dan IPv4, lihatMembatasi akses ke suatu Wilayah tertentu.

Anda dapat meninjau kebijakan pengguna IAM menggunakan konsol IAM di. https://console.aws.amazon.com/iam/ Untuk informasi lebih lanjut tentang IAM, lihat Panduan Pengguna IAM. Untuk informasi tentang mengedit S3 pada kebijakan bucket Outposts, lihat. Menambahkan atau mengedit kebijakan bucket untuk bucket Amazon S3 di Outposts

Menguji kompatibilitas alamat IP

Jika Anda menggunakan instance Linux atau Unix, atau platform macOS X, Anda dapat menguji akses Anda ke titik akhir tumpukan ganda. IPv6 Misalnya, untuk menguji koneksi ke Amazon S3 pada titik akhir Outposts, gunakan perintah: IPv6 `dig`

```
dig s3-outposts.us-west-2.api.aws AAAA +short
```

Jika titik akhir dual-stack Anda melalui IPv6 jaringan diatur dengan benar, `dig` perintah mengembalikan alamat yang terhubung. IPv6 Misalnya:

```
dig s3-outposts.us-west-2.api.aws AAAA +short

2600:1f14:2588:4800:b3a9:1460:159f:ebce

2600:1f14:2588:4802:6df6:c1fd:ef8a:fc76

2600:1f14:2588:4801:d802:8ccf:4e04:817
```

Menggunakan IPv6 dengan AWS PrivateLink

S3 di Outposts mendukung protokol AWS PrivateLink untuk layanan IPv6 dan titik akhir. Dengan AWS PrivateLink dukungan IPv6 protokol, Anda dapat terhubung ke titik akhir layanan dalam VPC IPv6 melalui jaringan, baik dari koneksi lokal maupun pribadi lainnya. IPv6 Dukungan AWS PrivateLink untuk S3 di Outposts juga memungkinkan Anda untuk AWS PrivateLink berintegrasi dengan titik akhir dual-stack. Untuk langkah-langkah tentang cara mengaktifkan IPv6 AWS PrivateLink, lihat Mempercepat IPv6 adopsi Anda dengan AWS PrivateLink layanan dan titik akhir.

> **Note**
>
> Untuk memperbarui jenis alamat IP yang didukung dari IPv4 ke IPv6, lihat Memodifikasi jenis alamat IP yang didukung di Panduan AWS PrivateLink Pengguna.

Menggunakan IPv6 dengan AWS PrivateLink

Jika Anda menggunakan AWS PrivateLink dengan IPv6, Anda harus membuat IPv6 atau dual-stack antarmuka VPC endpoint. Untuk langkah-langkah umum tentang cara membuat titik akhir VPC menggunakan AWS Management Console, lihat Mengakses AWS layanan menggunakan titik akhir VPC antarmuka di Panduan Pengguna.AWS PrivateLink

AWS Management Console

Gunakan prosedur berikut untuk membuat titik akhir VPC antarmuka yang terhubung ke S3 di Outposts.

1. Masuk ke AWS Management Console dan buka konsol VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Titik Akhir.
3. Pilih Buat Titik Akhir.
4. Untuk kategori Layanan, pilih AWS layanan.
5. Untuk nama Layanan, pilih layanan S3 on Outposts (com.amazonaws.us-east-1.s3-outposts).
6. Untuk VPC, pilih VPC dari mana Anda akan mengakses S3 di Outposts.
7. Untuk Subnet, pilih satu subnet per Availability Zone dari mana Anda akan mengakses S3 di Outposts. Anda tidak dapat memilih beberapa subnet dari Availability Zone yang sama. Untuk setiap subnet yang Anda pilih, antarmuka jaringan endpoint baru dibuat. Secara default, alamat IP dari rentang alamat IP subnet ditetapkan ke antarmuka jaringan titik akhir. Untuk menunjuk alamat IP untuk antarmuka jaringan titik akhir, pilih Tentukan alamat IP dan masukkan alamat dari rentang IPv6 alamat subnet.
8. Untuk jenis alamat IP, pilih Dualstack. Tetapkan keduanya IPv4 dan IPv6 alamat ke antarmuka jaringan titik akhir Anda. Opsi ini didukung hanya jika semua subnet yang dipilih memiliki rentang keduanya IPv4 dan IPv6 alamat.
9. Untuk grup Keamanan, pilih grup keamanan untuk dikaitkan dengan antarmuka jaringan titik akhir untuk titik akhir VPC. Secara default, grup keamanan default dikaitkan dengan VPC.
10. Untuk Kebijakan, pilih Akses penuh untuk mengizinkan semua operasi oleh semua pengguna utama pada semua sumber daya melalui titik akhir VPC. Jika tidak, pilih Kustom untuk melampirkan kebijakan titik akhir VPC yang mengontrol izin yang dimiliki kepala sekolah untuk melakukan tindakan pada sumber daya melalui titik akhir VPC. Opsi ini hanya tersedia jika layanan mendukung kebijakan titik akhir VPC. Untuk informasi selengkapnya, lihat Kebijakan Endpoint.

11. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
12. Pilih Buat titik akhir.

Example — Kebijakan bucket S3 pada Outposts

Untuk mengizinkan S3 di Outposts berinteraksi dengan titik akhir VPC Anda, Anda kemudian dapat memperbarui kebijakan S3 on Outposts seperti ini:

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3-outposts:*",
            "Resource": "*",
            "Principal": "*"
        }
    ]
}
```

AWS CLI

> **Note**
>
> Untuk mengaktifkan IPv6 jaringan pada titik akhir VPC Anda, Anda harus `IPv6` mengatur `SupportedIpAddressType` filter untuk S3 di Outposts.

Contoh berikut menggunakan `create-vpc-endpoint` perintah untuk membuat endpoint antarmuka dual-stack baru.

```
aws ec2 create-vpc-endpoint \
--vpc-id vpc-12345678 \
--vpc-endpoint-type Interface \
--service-name com.amazonaws.us-east-1.s3-outposts \
--subnet-id subnet-12345678 \
--security-group-id sg-12345678 \
--ip-address-type dualstack \
--dns-options "DnsRecordIpType=dualstack"
```

Bergantung pada konfigurasi AWS PrivateLink layanan, koneksi titik akhir yang baru dibuat mungkin perlu diterima oleh penyedia layanan titik akhir VPC sebelum dapat digunakan. Untuk informasi selengkapnya, lihat Menerima dan menolak permintaan koneksi titik akhir di AWS PrivateLink Panduan Pengguna.

Contoh berikut menggunakan `modify-vpc-endpoint` perintah untuk memperbarui titik akhir VPC IPv -only ke titik akhir dual-stack. Titik akhir dual-stack memungkinkan akses ke jaringan dan jaringan. IPv4 IPv6

```
aws ec2 modify-vpc-endpoint \
--vpc-endpoint-id vpce-12345678 \
--add-subnet-ids subnet-12345678 \
--remove-subnet-ids subnet-12345678 \
--ip-address-type dualstack \
--dns-options "DnsRecordIpType=dualstack"
```

Untuk informasi selengkapnya tentang cara mengaktifkan IPv6 jaringan AWS PrivateLink, lihat Mempercepat IPv6 adopsi Anda dengan AWS PrivateLink layanan dan titik akhir.

Menggunakan S3 pada titik akhir dual-stack Outposts

S3 di Outposts dual-stack endpoint mendukung permintaan ke S3 pada bucket Outposts over dan. IPv6 IPv4 Bagian ini menjelaskan cara menggunakan S3 pada titik akhir dual-stack Outposts.

Topik

- S3 pada titik akhir tumpukan ganda Outposts
- Menggunakan titik akhir tumpukan ganda dari AWS CLI
- Menggunakan S3 pada titik akhir tumpukan ganda Outposts dari AWS SDKs

S3 pada titik akhir tumpukan ganda Outposts

Saat Anda membuat permintaan ke titik akhir dual-stack, URL bucket S3 on Outposts akan diselesaikan ke alamat atau alamat. IPv6 IPv4 Untuk informasi selengkapnya tentang mengakses bucket S3 on Outposts, lihat. IPv6 Membuat permintaan ke S3 di Outposts over IPv6

Untuk mengakses bucket S3 di Outposts melalui titik akhir dual-stack, gunakan nama endpoint gaya jalur. S3 di Outposts hanya mendukung nama titik akhir tumpukan ganda Regional, yang berarti Anda harus menentukan Wilayah sebagai bagian dari nama.

Untuk FIPs titik akhir gaya jalur tumpukan ganda, gunakan konvensi penamaan berikut:

```
s3-outposts-fips.region.api.aws
```

Untuk titik akhir non-FIPS dual-stack, gunakan konvensi penamaan berikut:

```
s3-outposts.region.api.aws
```

> **Note**
>
> Nama titik akhir bergaya host virtual tidak didukung di S3 di Outposts.

Menggunakan titik akhir tumpukan ganda dari AWS CLI

Bagian ini memberikan contoh AWS CLI perintah yang digunakan untuk membuat permintaan ke endpoint dual-stack. Untuk petunjuk tentang pengaturan AWS CLI, lihatMemulai dengan menggunakan AWS CLI dan SDK for Java.

Anda menetapkan nilai konfigurasi `use_dualstack_endpoint` ke `true` dalam profil dalam AWS Config file Anda untuk mengarahkan semua permintaan Amazon S3 yang dibuat oleh `s3api` AWS CLI perintah `s3` dan ke titik akhir tumpukan ganda untuk Wilayah yang ditentukan. Anda menentukan Wilayah dalam file konfigurasi atau dalam perintah menggunakan `--region` opsi.

Saat menggunakan titik akhir tumpukan ganda dengan AWS CLI, hanya gaya `path` pengalamatan yang didukung. Gaya pengalamatan, yang diatur dalam file konfigurasi, menentukan apakah nama bucket ada di nama host atau di URL. Untuk informasi selengkapnya, lihat s3outposts di AWS CLI Panduan Pengguna.

Untuk menggunakan titik akhir tumpukan ganda melalui AWS CLI, gunakan `--endpoint-url` parameter dengan `https://s3-outposts-fips.region.api.aws` titik akhir `http://s3.dualstack.region.amazonaws.com` atau untuk perintah apa pun atau. `s3control` `s3outposts`

Misalnya:

```
$  aws s3control list-regional-buckets --endpoint-url https://s3-
outposts.region.api.aws
```

Menggunakan S3 pada titik akhir tumpukan ganda Outposts dari AWS SDKs

Bagian ini memberikan contoh cara mengakses endpoint dual-stack dengan menggunakan. AWS SDKs

Contoh titik akhir tumpukan ganda AWS SDK for Java 2.x

Contoh berikut menunjukkan cara menggunakan `S3OutpostsClient` kelas `S3ControlClient` dan untuk mengaktifkan titik akhir dual-stack saat membuat klien S3 di Outposts menggunakan file. AWS SDK for Java 2.x Untuk petunjuk tentang membuat dan menguji contoh Java yang berfungsi untuk Amazon S3 di Outposts, lihat. Memulai dengan menggunakan AWS CLI dan SDK for Java

Example — Buat **`S3ControlClient`** kelas dengan titik akhir dual-stack diaktifkan

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.s3control.S3ControlClient;
import software.amazon.awssdk.services.s3control.model.ListRegionalBucketsRequest;
import software.amazon.awssdk.services.s3control.model.ListRegionalBucketsResponse;
import software.amazon.awssdk.services.s3control.model.S3ControlException;


public class DualStackEndpointsExample1 {

    public static void main(String[] args) {
        Region clientRegion = Region.of("us-east-1");
        String accountId = "111122223333";
        String navyId = "9876543210";

        try {
            // Create an S3ControlClient with dual-stack endpoints enabled.
            S3ControlClient s3ControlClient = S3ControlClient.builder()
                                                             .region(clientRegion)
                                                             .dualstackEnabled(true)
                                                             .build();
            ListRegionalBucketsRequest listRegionalBucketsRequest =
 ListRegionalBucketsRequest.builder()

       .accountId(accountId)

       .outpostId(navyId)
```

```
       .build();

            ListRegionalBucketsResponse listBuckets =
 s3ControlClient.listRegionalBuckets(listRegionalBucketsRequest);
            System.out.printf("ListRegionalBuckets Response: %s%n",
 listBuckets.toString());
       } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 on Outposts
 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
       }
       catch (S3ControlException e) {
            // Unknown exceptions will be thrown as an instance of this type.
            e.printStackTrace();
       } catch (SdkClientException e) {
            // Amazon S3 on Outposts couldn't be contacted for a response, or the
 client
            // couldn't parse the response from Amazon S3 on Outposts.
            e.printStackTrace();
       }
   }
}
```

Example — Buat **`S3OutpostsClient`** dengan titik akhir dual-stack diaktifkan

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.s3outposts.S3OutpostsClient;
import software.amazon.awssdk.services.s3outposts.model.ListEndpointsRequest;
import software.amazon.awssdk.services.s3outposts.model.ListEndpointsResponse;
import software.amazon.awssdk.services.s3outposts.model.S3OutpostsException;


public class DualStackEndpointsExample2 {

    public static void main(String[] args) {
        Region clientRegion = Region.of("us-east-1");

        try {
            // Create an S3OutpostsClient with dual-stack endpoints enabled.
```

```
            S3OutpostsClient s3OutpostsClient = S3OutpostsClient.builder()
                                                                .region(clientRegion)
                                                                .dualstackEnabled(true)
                                                                .build();
            ListEndpointsRequest listEndpointsRequest =
 ListEndpointsRequest.builder().build();

            ListEndpointsResponse listEndpoints =
 s3OutpostsClient.listEndpoints(listEndpointsRequest);
            System.out.printf("ListEndpoints Response: %s%n",
 listEndpoints.toString());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 on Outposts
 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        }
        catch (S3OutpostsException e) {
            // Unknown exceptions will be thrown as an instance of this type.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 on Outposts couldn't be contacted for a response, or the
 client
            // couldn't parse the response from Amazon S3 on Outposts.
            e.printStackTrace();
        }
    }
}
```

Jika Anda menggunakan AWS SDK for Java 2.x pada Windows, Anda mungkin harus mengatur properti Java virtual machine (JVM) berikut:

```
java.net.preferIPv6Addresses=true
```

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.