Panduan Pengguna

AWS Ground Station

AWS Ground Station: Panduan Pengguna

Table of Contents

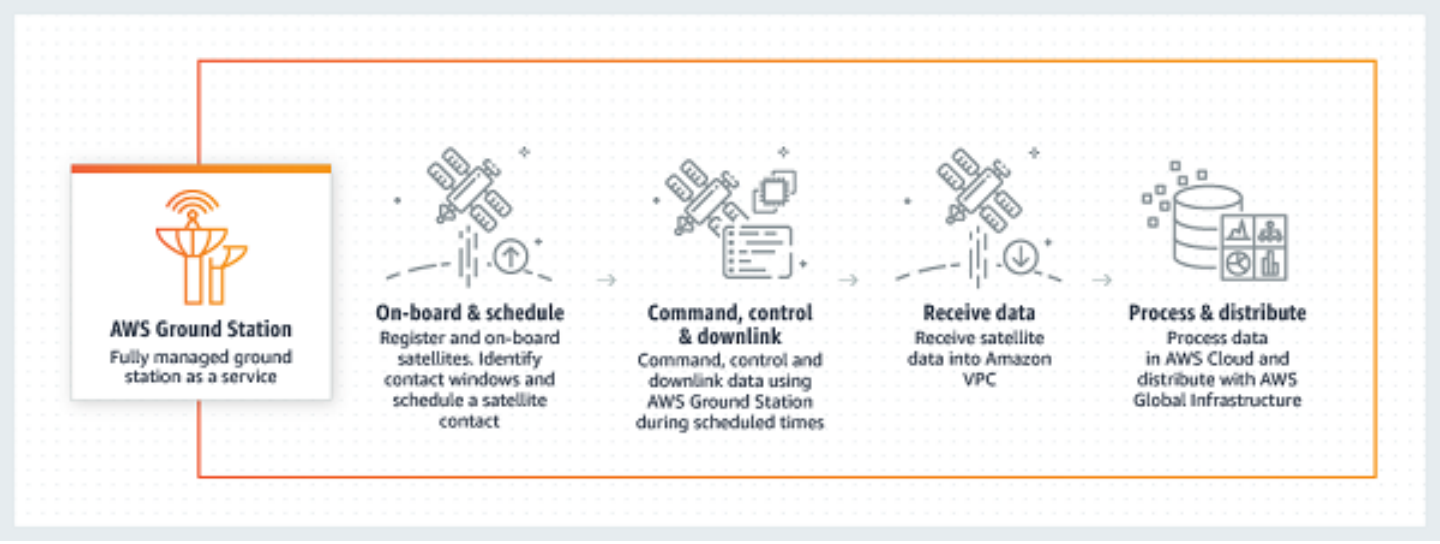
Apa itu AWS Ground Station?

AWS Ground Station adalah layanan yang dikelola sepenuhnya yang menyediakan komunikasi satelit yang aman, cepat, dan dapat diprediksi di seluruh infrastruktur global. Dengan AWS Ground Station, Anda tidak lagi harus membangun, mengelola, atau menskalakan infrastruktur stasiun bumi Anda sendiri. AWS Ground Station memungkinkan Anda untuk fokus pada inovasi dan bereksperimen dengan cepat dengan aplikasi baru yang menyerap data satelit, daripada menghabiskan sumber daya untuk membangun, mengoperasikan, dan menskalakan stasiun bumi Anda sendiri.

Dengan menggunakan jaringan serat global dengan latensi rendah dan bandwidth tinggi AWS, Anda dapat mulai memproses data satelit Anda dalam hitungan detik setelah penerimaan di sistem antena. Ini memungkinkan Anda untuk mengubah data mentah menjadi informasi yang diproses atau pengetahuan yang dianalisis dalam hitungan detik.

Untuk organisasi dengan persyaratan khusus, AWS Ground Station juga menawarkan AWS Ground Station Antena Khusus — sistem antena yang dibuat khusus yang AWS mengelola atas nama Anda, menyediakan akses khusus ke antena yang dibangun sesuai spesifikasi Anda.

Kasus penggunaan umum



AWS Ground Station memungkinkan Anda untuk berkomunikasi dengan satelit Anda dua arah dan mendukung kasus penggunaan berikut:

- Data downlink — Menerima data dari satelit Anda, mentransmisikan frekuensi X-band dan S-band, dikirimkan ke instans Amazon EC2 secara real-time (format VITA-49), atau langsung ke bucket Amazon S3 di akun Anda (format PCAP). Selain itu, untuk satelit yang menggunakan skema modulasi dan pengkodean yang didukung, Anda dapat memilih antara menerima data yang didemodulasi dan diterjemahkan, atau sampel frekuensi menengah digital mentah (DiGIF) (format VITA-49).
- Uplink data — Kirim data dan perintah ke satelit Anda, yang menerima frekuensi S-band, dengan mengirimkan data DiGIF (format VITA-49) untuk ditransmisikan oleh. AWS Ground Station
- Uplink echo — Validasi perintah yang dikirim ke pesawat ruang angkasa Anda, dan lakukan tugas-tugas lanjutan lainnya, dengan menerima sinyal yang ditransmisikan pada antena yang ditempatkan bersama secara fisik.
- Software Defined Radio (SDR) /Front End Processor (FEP) - Gunakan SDR and/or FEP Anda yang ada, yang mampu berjalan pada instans Amazon EC2, untuk memproses data Anda secara real-time ke bentuk gelombang yang ada, dan menghasilkan produk data send/receive Anda.
- Telemetri, Pelacakan, dan Perintah (TT&C) — Lakukan TT&C menggunakan kombinasi kasus penggunaan yang terdaftar sebelumnya untuk mengelola armada satelit Anda.
- Pengiriman Data Lintas Wilayah — Mengoperasikan beberapa kontak simultan menggunakan AWS Ground Station jaringan antena global dari satu Wilayah AWS.
- Digital twin — Uji penjadwalan, verifikasi konfigurasi, dan penanganan kesalahan yang tepat dengan biaya yang lebih rendah tanpa menggunakan kapasitas antena produksi.

Langkah selanjutnya

Kami menyarankan Anda untuk memulai dengan membaca bagian berikut:

- Untuk mempelajari AWS Ground Station konsep-konsep penting, lihatBagaimana cara AWS Ground Station kerja.
- Untuk mempelajari cara menyiapkan akun dan sumber daya yang akan digunakan AWS Ground Station, lihatMemulai.
- Untuk menggunakan secara terprogram AWS Ground Station, silakan merujuk ke Referensi AWS Ground Station API. Referensi API menjelaskan semua operasi API AWS Ground Station secara detail. Ini juga menyediakan permintaan sampel, tanggapan, dan kesalahan untuk protokol layanan web yang didukung. Anda dapat menggunakan AWS CLI, atau AWS SDK, dalam bahasa pilihan Anda, untuk menulis kode yang berinteraksi dengannya. AWS Ground Station

Bagaimana cara AWS Ground Station kerja

AWS Ground Station mengoperasikan antena berbasis darat untuk memfasilitasi komunikasi dengan satelit Anda. Karakteristik fisik dari apa yang dapat dilakukan antena diabstraksikan dan disebut sebagai kemampuan. Lokasi fisik antena beserta kemampuannya saat ini dapat direferensikan di AWS Ground Station Lokasi bagian tersebut. Silakan hubungi kami melalui AWS Support Center Consolejika kasus penggunaan Anda memerlukan kemampuan tambahan, penawaran lokasi tambahan, atau lokasi antena yang lebih tepat.

Untuk menggunakan salah satu AWS Ground Station antena, Anda harus memesan waktu di lokasi tertentu. Reservasi ini disebut sebagai kontak. Untuk berhasil menjadwalkan kontak, AWS Ground Station memerlukan data tambahan untuk memastikan keberhasilannya.

- Satelit Anda harus onboard ke satu atau beberapa lokasi — Ini memastikan Anda memiliki persetujuan untuk mengoperasikan berbagai kemampuan di lokasi yang diminta.
- Satelit Anda harus memiliki ephemeris yang valid — Ini memastikan antena memiliki garis pandang dan dapat secara akurat menunjuk ke satelit Anda selama kontak.
- Anda harus memiliki profil misi yang valid — Ini memungkinkan Anda untuk menyesuaikan bagaimana kontak ini akan berperilaku termasuk bagaimana Anda akan menerima dan mengirim data ke satelit Anda. Anda dapat menggunakan beberapa profil misi untuk kendaraan yang sama untuk membuat kontak yang berbeda agar sesuai dengan postur operasi atau skenario yang berbeda yang Anda temui.

Orientasi satelit

Memasukkan satelit ke dalam AWS Ground Station adalah proses multistep yang melibatkan pengumpulan data, validasi teknis, lisensi spektrum, dengan integrasi dan pengujian. Bagian orientasi satelit dari panduan ini akan memandu Anda melalui proses ini.

Komposisi profil misi

Informasi frekuensi satelit, informasi pesawat data, dan detail lainnya dienkapsulasi ke dalam profil misi. Profil misi adalah kumpulan komponen konfigurasi. Ini memungkinkan Anda untuk menggunakan kembali komponen konfigurasi di berbagai profil misi yang sesuai dengan kasus penggunaan Anda. Karena profil misi tidak secara langsung merujuk satelit individu, tetapi hanya

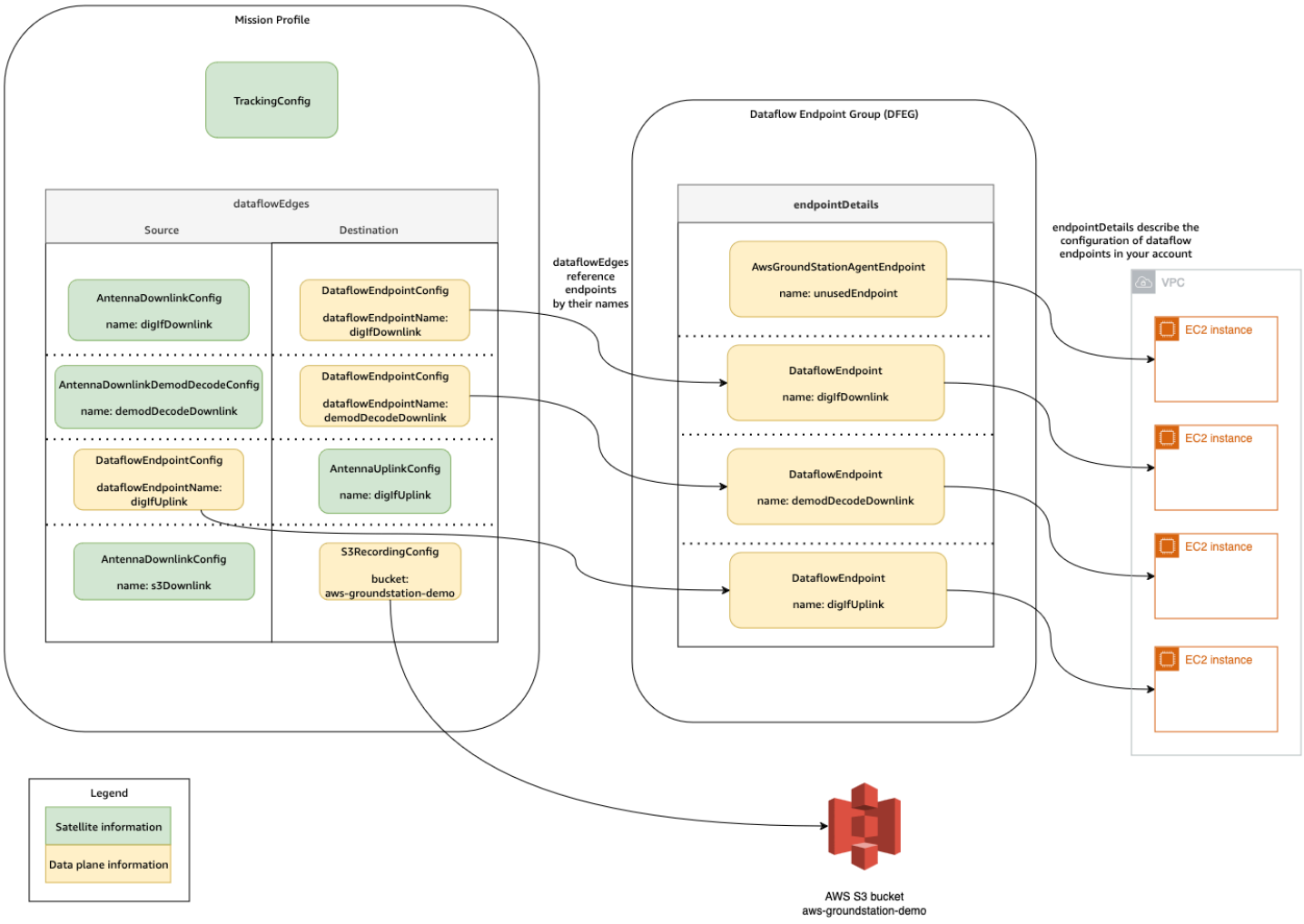
memiliki informasi tentang kemampuan teknis mereka, profil misi juga dapat digunakan kembali oleh beberapa satelit yang memiliki konfigurasi yang sama.

Profil misi yang valid akan memiliki konfigurasi pelacakan dan satu atau beberapa aliran data. Konfigurasi pelacakan akan menentukan preferensi Anda untuk melacak selama kontak. Setiap pasangan konfigurasi dalam aliran data menetapkan sumber dan tujuan. Bergantung pada satelit Anda dan mode operasionalnya, jumlah aliran data yang tepat akan bervariasi dalam profil misi untuk mewakili jalur komunikasi uplink dan downlink Anda serta aspek pemrosesan data apa pun.

- Untuk informasi selengkapnya tentang mengonfigurasi VPC Amazon, Amazon S3, dan sumber daya EC2 Amazon yang akan digunakan selama kontak, lihat. Bekerja dengan aliran data
- Untuk detail tentang bagaimana setiap konfigurasi berperilaku, lihat. Gunakan AWS Ground Station Konfigurasi
- Untuk detail spesifik tentang semua parameter yang diharapkan, lihatGunakan Profil AWS Ground Station Misi.
- Untuk contoh tentang bagaimana berbagai profil misi dapat dibuat untuk mendukung kasus penggunaan Anda, lihatContoh konfigurasi profil misi.

Diagram berikut menunjukkan contoh profil misi dan sumber daya tambahan yang dibutuhkan. Perhatikan bahwa contoh menunjukkan titik akhir aliran data yang tidak diperlukan untuk profil misi ini, bernama UnusedEndPoint, untuk menunjukkan fleksibilitas. Contoh ini mendukung aliran data berikut:

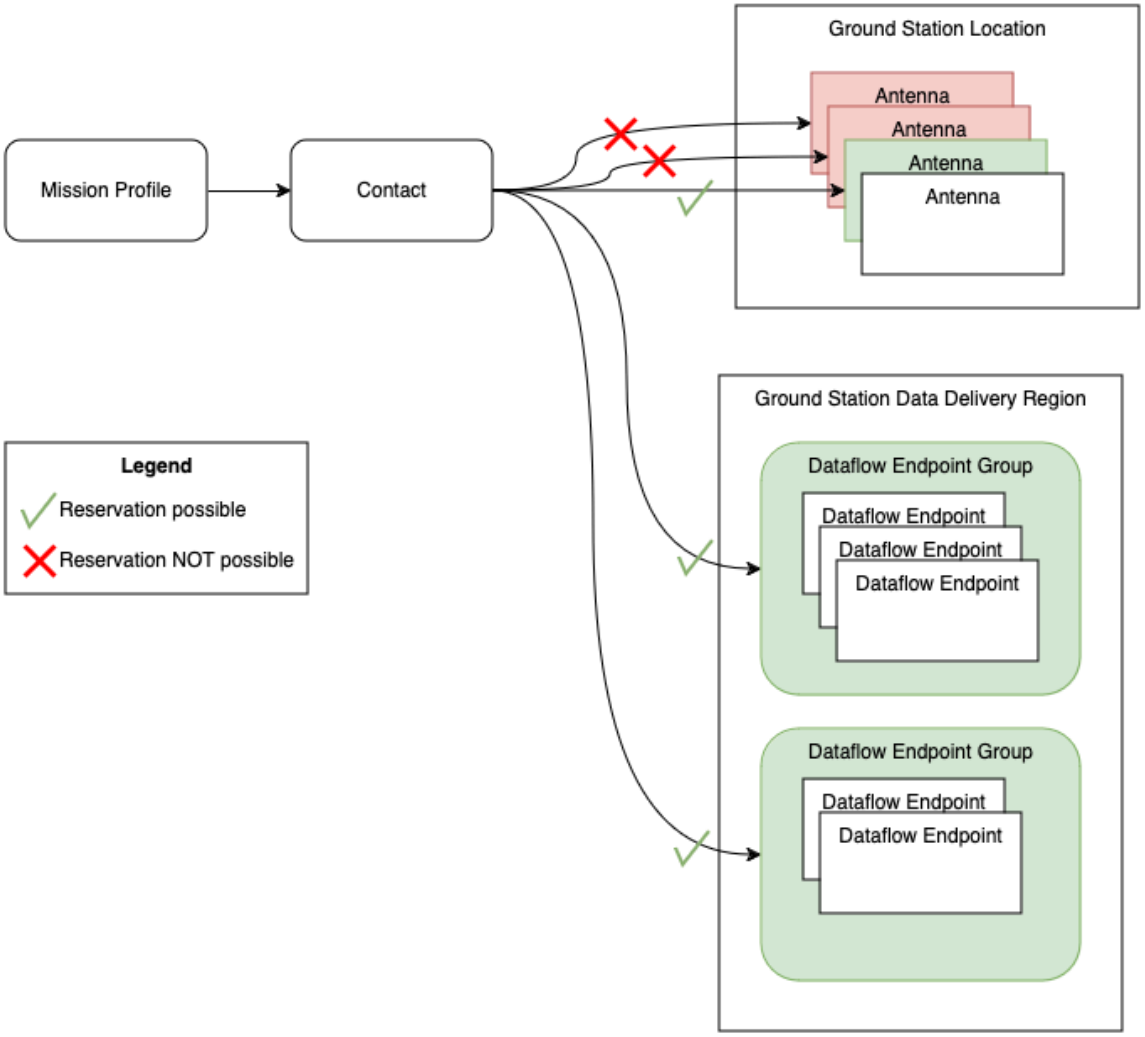
- Downlink sinkron data frekuensi menengah digital ke EC2 instans Amazon yang Anda kelola. Dilambangkan dengan nama. digIfDownlink
- Downlink asinkron data frekuensi menengah digital ke bucket Amazon S3. Dilambangkan dengan nama ember. aws-groundstation-demo
- Downlink sinkron data yang didemodulasi dan diterjemahkan ke instans Amazon EC2 yang Anda kelola. Dilambangkan dengan nama. demodDecodeDownlink
- Uplink sinkron data dari EC2 instans Amazon yang Anda kelola ke antena terkelola. AWS Ground Station Dilambangkan dengan nama. digIfUplink



Penjadwalan kontak

Dengan profil misi yang valid, Anda dapat meminta kontak dengan satelit onboard Anda. Permintaan reservasi kontak bersifat asinkron untuk memberikan waktu bagi layanan antena global untuk mencapai jadwal yang konsisten di semua AWS Wilayah yang terlibat. Selama proses ini, berbagai antena di lokasi stasiun bumi yang diminta dievaluasi untuk menentukan apakah tersedia dan mampu memproses kontak. Selama proses ini, titik akhir aliran data Anda yang dikonfigurasi juga dievaluasi untuk menentukan ketersediaannya. Sementara evaluasi ini terjadi, status kontak akan dalam PENJADWALAN.

Proses penjadwalan asinkron ini akan selesai dalam waktu lima menit setelah permintaan, tetapi biasanya selesai dalam satu menit. Harap tinjau pemantauan berbasis acara Otomatisasi AWS Ground Station dengan Acara selama waktu penjadwalan.



Kontak yang dapat dilakukan dan memiliki ketersediaan menghasilkan kontak TERJADWAL. Dengan kontak terjadwal, sumber daya yang diperlukan untuk melakukan kontak Anda telah dipesan di seluruh Wilayah AWS yang diperlukan sebagaimana ditentukan oleh profil misi Anda. Kontak yang tidak dapat dilakukan, atau memiliki bagian yang tidak tersedia akan menghasilkan kontak FAILED_TO_SCHEDULE. Lihat Memecahkan masalah kontak FAILED_TO_SCHEDULE detail debugging.

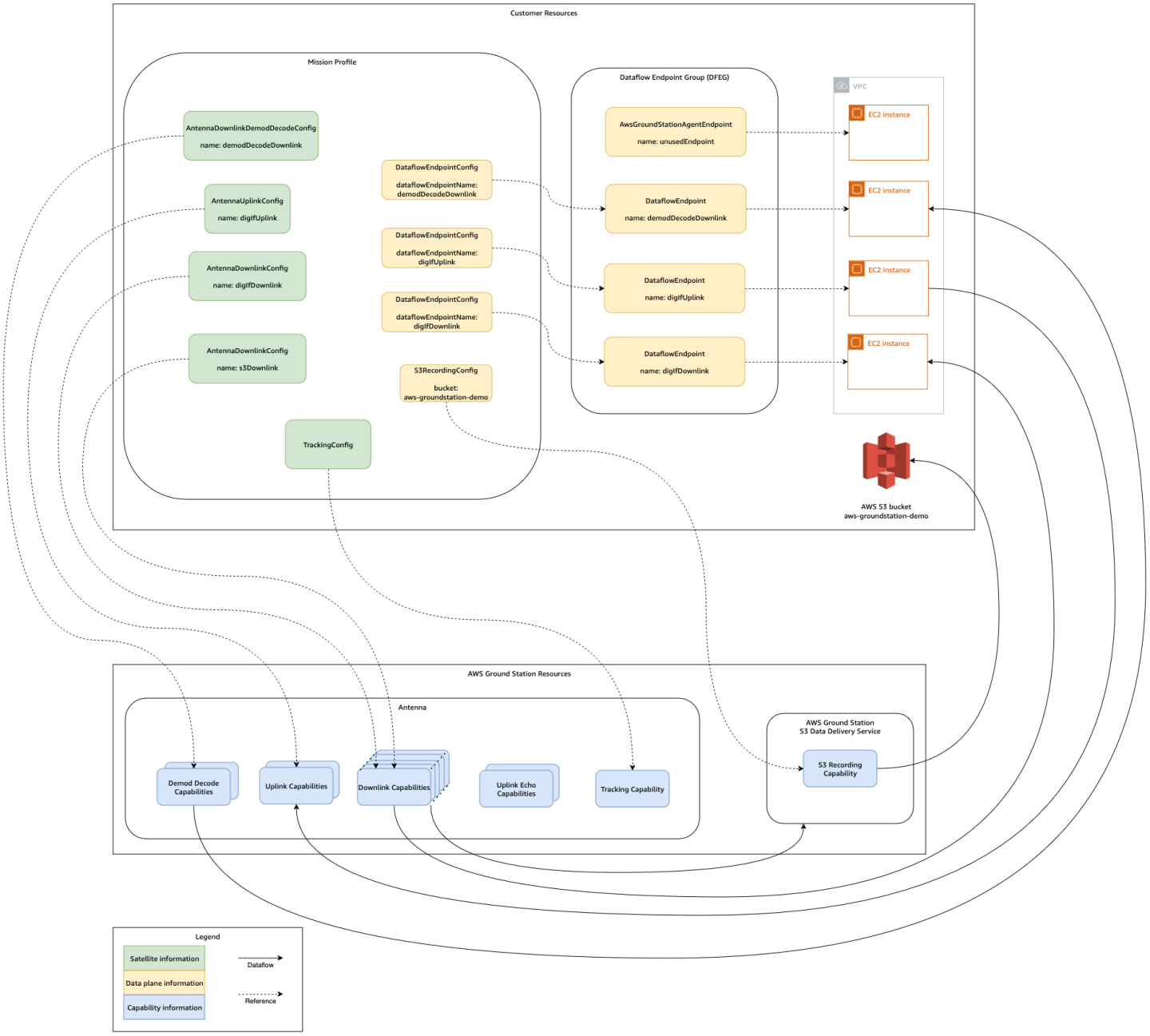
Eksekusi kontak

AWS Ground Station akan secara otomatis mengatur sumber daya yang dikelola AWS Anda selama reservasi kontak Anda. Jika berlaku, Anda bertanggung jawab untuk mengatur EC2 sumber daya yang ditentukan oleh profil misi Anda sebagai titik akhir aliran data. AWS Ground Station menyediakan AWS EventBridge Events untuk mengotomatiskan orkestrasi sumber daya Anda untuk mengurangi biaya. Lihat Otomatisasi AWS Ground Station dengan Acara untuk detail selengkapnya.

Selama kontak, telemetri tentang kinerja kontak Anda dikirimkan ke AWS. CloudWatch Untuk informasi tentang cara memantau kontak Anda selama eksekusi, silakan lihatMemahami pemantauan dengan AWS Ground Station.

Diagram berikut melanjutkan contoh sebelumnya dengan menunjukkan sumber daya yang sama diatur selama kontak.

> **Note**
>
> Tidak semua kemampuan antena digunakan dalam contoh ini. Misalnya, ada lebih dari selusin kemampuan downlink antena yang tersedia di setiap antena yang mendukung beberapa frekuensi dan polarisasi. Untuk detail lebih lanjut tentang jumlah setiap jenis kemampuan yang tersedia dari AWS Ground Station antena, dan frekuensi dan polarisasi yang didukung, lihat. AWS Ground Station Kemampuan Situs



Di akhir kontak Anda, AWS Ground Station akan menilai kinerja kontak Anda dan akan menentukan status kontak akhir. Kontak di mana tidak ada kesalahan yang terdeteksi akan menghasilkan status kontak LENGKAP. Kontak di mana kesalahan layanan telah menyebabkan masalah pengiriman data selama kontak akan menghasilkan AWS_FAILEDstatus. Kontak di mana kesalahan klien atau pengguna menyebabkan masalah pengiriman data selama kontak akan menghasilkan status GAGAL. Kesalahan di luar waktu kontak, yaitu selama pre-pass atau post-pass, tidak diperhitungkan selama ajudikasi.

Untuk informasi selengkapnya, lihat Memahami siklus hidup kontak.

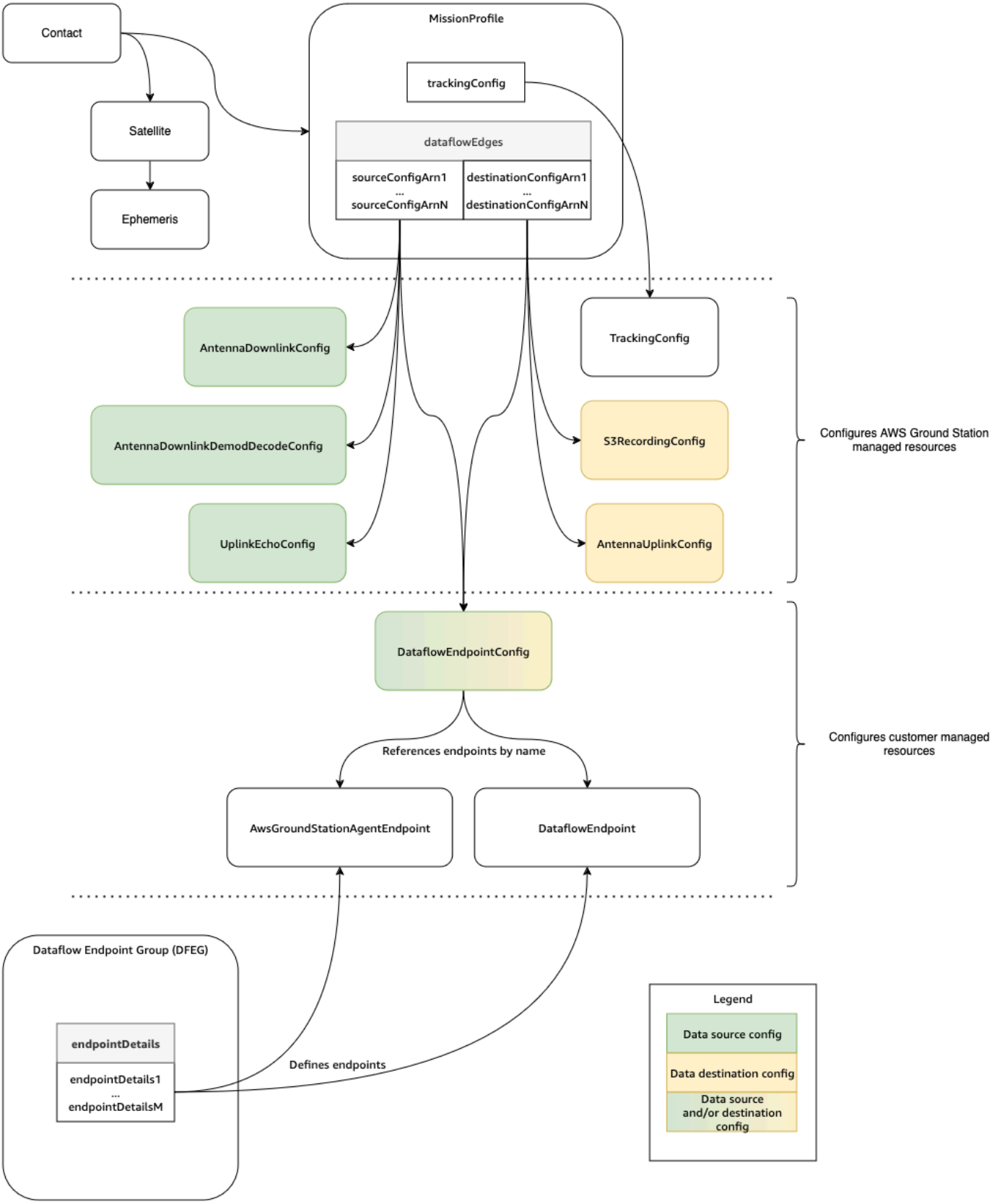
Kembar digital

Fitur kembar digital untuk AWS Ground Station memungkinkan Anda menjadwalkan kontak dengan lokasi stasiun bumi virtual. Stasiun bumi virtual ini adalah replika yang tepat dari stasiun bumi produksi termasuk kemampuan antena, masker situs, dan koordinat GPS yang sebenarnya. Fitur kembar digital memungkinkan Anda untuk menguji alur kerja orkestrasi kontak Anda untuk sebagian kecil dari biaya dibandingkan dengan stasiun bumi produksi. Lihat Gunakan AWS Ground Station fitur kembar digital untuk informasi selengkapnya.

Memahami komponen AWS Ground Station Inti

Bagian ini memberikan definisi terperinci untuk komponen inti AWS Ground Station.

Diagram berikut menunjukkan komponen inti AWS Ground Station dan bagaimana mereka berhubungan satu sama lain. Panah menunjukkan arah dependensi antar komponen, di mana setiap komponen menunjuk ke dependensinya.


Contact
Satellite
Ephemeris
MissionProfile
trackingConfig
dataflowEdges
sourceConfigArn1
...
sourceConfigArnN
destinationConfigArn1
...
destinationConfigArnN
AntennaDownlinkConfig
AntennaDownlinkDemodDecodeConfig
UplinkEchoConfig
TrackingConfig
S3RecordingConfig
AntennaUplinkConfig
Configures AWS Ground Station managed resources
DataflowEndpointConfig
References endpoints by name
AwsGroundStationAgentEndpoint
DataflowEndpoint
Configures customer managed resources
Dataflow Endpoint Group (DFEG)
endpointDetails
endpointDetails1
...
endpointDetailsM
Defines endpoints
Legend
Data source config
Data destination config
Data source and/or destination config

Topik berikut menjelaskan komponen AWS Ground Station inti secara rinci.

Topik

Gunakan Profil AWS Ground Station Misi

Profil misi berisi konfigurasi dan parameter untuk bagaimana kontak dijalankan. Ketika Anda memesan kontak atau mencari kontak yang tersedia, Anda menyediakan profil misi yang ingin Anda gunakan. Profil misi menyatukan semua konfigurasi Anda dan menentukan bagaimana antena akan dikonfigurasi dan ke mana data akan pergi selama kontak Anda.

Profil misi dapat dibagikan di seluruh satelit yang memiliki karakteristik radio yang sama. Anda dapat membuat grup titik akhir aliran data tambahan untuk mengikat kontak simultan maksimum yang ingin Anda lakukan untuk konstelasi Anda.

Konfigurasi pelacakan ditentukan sebagai bidang unik dalam profil misi. Konfigurasi pelacakan digunakan untuk menentukan preferensi Anda untuk menggunakan pelacakan program dan pelacakan otomatis selama kontak Anda. Untuk informasi selengkapnya, lihat Melacak Config.

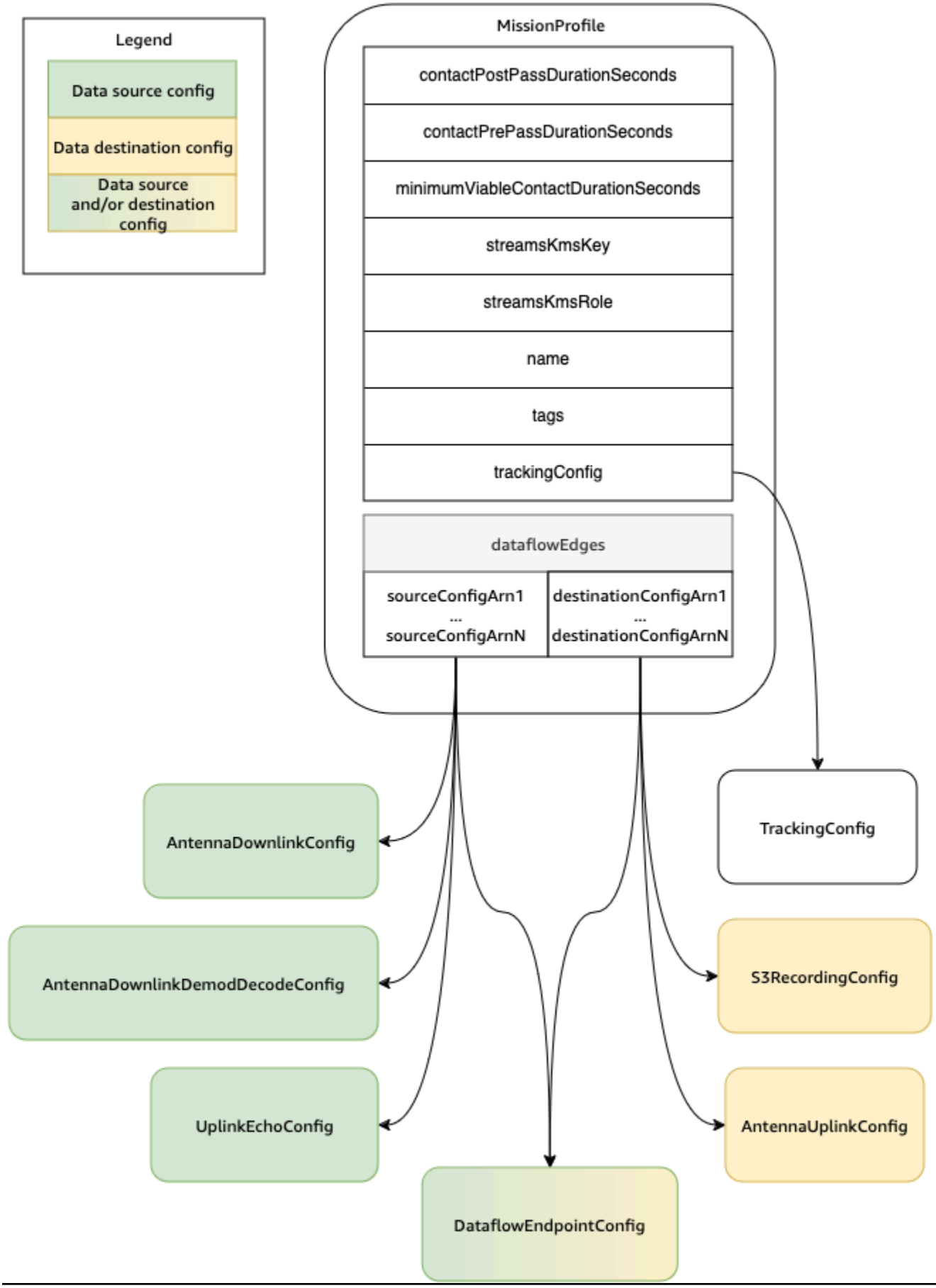
Semua konfigurasi lainnya terkandung di `dataflowEdges` bidang profil misi. Konfigurasi ini dapat dianggap sebagai node aliran data yang masing-masing mewakili sumber daya AWS Ground Station terkelola yang dapat mengirim atau menerima data dan konfigurasi terkait. `dataflowEdges`Bidang menentukan node aliran data sumber dan tujuan (konfigurasi) mana yang diperlukan. Satu tepi aliran data adalah daftar dua konfigurasi Amazon Resource Names ARNs () —yang pertama adalah konfigurasi sumber dan yang kedua adalah konfigurasi tujuan. Dengan menentukan tepi aliran data antara dua konfigurasi, Anda memberi tahu AWS Ground Station dari mana dan ke mana data harus mengalir selama kontak. Untuk informasi selengkapnya, lihat Gunakan AWS Ground Station Konfigurasi.

Itu `contactPrePassDurationSeconds` dan `contactPostPassDurationSeconds` memungkinkan Anda untuk menentukan waktu relatif terhadap kontak di mana Anda akan menerima pemberitahuan CloudWatch Acara. Untuk jadwal acara yang terkait dengan kontak Anda, silakan bacaMemahami siklus hidup kontak.

`name`Bidang profil misi membantu membedakan antara profil misi yang Anda buat.

`streamsKmsRole`Dan `streamsKmsKey` digunakan untuk menentukan enkripsi yang digunakan oleh AWS Ground Station untuk pengiriman data Anda dengan AWS Ground Station Agen. Silakan lihatEnkripsi data selama transit untuk AWS Ground Station.

`telemetrySinkConfigArn`Bidang ini opsional dan memungkinkan Anda mengaktifkan AWS Ground Station telemetri selama kontak. Saat ditentukan, AWS Ground Station mengalirkan data telemetri mendekati waktu nyata ke akun Anda selama eksekusi kontak Anda. Untuk informasi selengkapnya tentang mengonfigurasi dan menggunakan telemetri, lihat. Bekerja dengan telemetri


Legend
Data source config
Data destination config
Data source and/or destination config
MissionProfile
contactPostPassDurationSeconds
contactPrePassDurationSeconds
minimumViableContactDurationSeconds
streamsKmsKey
streamsKmsRole
name
tags
trackingConfig
dataflowEdges
sourceConfigArn1 ... sourceConfigArnN
destinationConfigArn1 ... destinationConfigArnN
AntennaDownlinkConfig
AntennaDownlinkDemodDecodeConfig
UplinkEchoConfig
TrackingConfig
S3RecordingConfig
AntennaUplinkConfig
DataflowEndpointConfig

Daftar lengkap parameter dan contoh disertakan pada dokumentasi berikut.

- AWS::GroundStation::MissionProfile CloudFormation jenis sumber daya

Gunakan AWS Ground Station Konfigurasi

Konfigurasi adalah sumber daya yang AWS Ground Station digunakan untuk menentukan parameter untuk setiap aspek kontak Anda. Tambahkan konfigurasi yang Anda inginkan ke profil misi, dan kemudian profil misi itu akan digunakan saat menjalankan kontak. Anda dapat menentukan beberapa jenis konfigurasi yang berbeda. Konfigurasi dapat dikelompokkan menjadi tiga kategori:

- Melacak konfigurasi
- Konfigurasi aliran data
- Konfigurasi telemetri

A TrackingConfigadalah satu-satunya jenis konfigurasi pelacakan. Ini digunakan untuk mengkonfigurasi pengaturan autotrack antena selama kontak, dan diperlukan dalam profil misi.

Konfigurasi yang dapat digunakan dalam aliran data profil misi dapat dianggap sebagai node aliran data yang masing-masing mewakili sumber daya AWS Ground Station terkelola yang dapat mengirim atau menerima data. Profil misi memerlukan setidaknya satu pasang konfigurasi ini, dengan satu mewakili sumber data, dan satu mewakili tujuan. Konfigurasi ini dirangkum dalam tabel berikut.

Nama Config	Sumber/tujuan aliran data
AntennaDownlinkConfig	Sumber
AntennaDownlinkDemodDecodeConfig	Sumber
UplinkEchoConfig	Sumber
S3 RecordingConfig	Tujuan
AntennaUplinkConfig	Tujuan
DataflowEndpointConfig	Sumber and/or Tujuan

A TelemetrySinkConfigadalah satu-satunya jenis konfigurasi telemetri. Ini digunakan untuk mengonfigurasi di mana data telemetri akan dikirimkan selama kontak, dan opsional dalam profil misi. Ketika disertakan, AWS Ground Station streaming dekat telemetri real-time ke akun Anda selama eksekusi kontak Anda.

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi menggunakan CloudFormation, API AWS Command Line Interface, atau AWS Ground Station API. Tautan ke dokumentasi untuk jenis konfigurasi tertentu juga disediakan di bawah ini.

- AWS::GroundStation::Config CloudFormation jenis sumber daya
- Referensi Config AWS CLI
- Referensi API Config

Melacak Config

Anda dapat menggunakan konfigurasi pelacakan di profil misi untuk menentukan apakah autotrack harus diaktifkan selama kontak Anda. Konfigurasi ini memiliki satu parameter:`autotrack`. `autotrack`Parameter dapat memiliki nilai-nilai berikut:

- `REQUIRED`- Autotrack diperlukan untuk kontak Anda.
- `PREFERRED`- Autotrack lebih disukai untuk kontak, tetapi kontak masih dapat dieksekusi tanpa autotrack.
- `REMOVED`- Tidak ada autotrack yang harus digunakan untuk kontak Anda.

AWS Ground Station akan menggunakan pelacakan terprogram yang akan menunjuk berdasarkan ephemeris Anda saat autotrack tidak digunakan. Silakan referensi Memahami caranya AWS Ground Station menggunakan ephemerides untuk detail tentang bagaimana ephemeris dibangun.

Autotrack akan menggunakan pelacakan program sampai sinyal yang diharapkan ditemukan. Setelah itu terjadi, ia akan terus melacak berdasarkan kekuatan sinyal.

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi pelacakan menggunakan CloudFormation, API AWS Command Line Interface, atau AWS Ground Station API.

- AWS::GroundStation::Config TrackingConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`trackingConfig -> (structure)`)

- TrackingConfig Referensi API

Konfigurasi Downlink Antena

Anda dapat menggunakan konfigurasi downlink antena untuk mengonfigurasi antena untuk downlink selama kontak Anda. Mereka terdiri dari konfigurasi spektrum yang menentukan frekuensi, bandwidth, dan polarisasi yang harus digunakan selama kontak downlink Anda.

Konfigurasi ini merupakan node sumber dalam aliran data. Ini bertanggung jawab untuk mendigitalkan data frekuensi radio. Data yang dialirkan dari node ini akan mengikuti Data/IP Format Sinyal. Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Jika kasus penggunaan downlink Anda memerlukan demodulasi atau decoding, lihat. Antena Downlink Demod Decode Config

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi downlink antena menggunakan CloudFormation, API AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config AntennaDownlinkConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`antennaDownlinkConfig -> (structure)`)
- AntennaDownlinkConfig Referensi API

Antena Downlink Demod Decode Config

Konfigurasi decode demod downlink antena adalah jenis konfigurasi yang lebih kompleks dan dapat disesuaikan yang dapat Anda gunakan untuk menjalankan kontak downlink dengan decoding demodulasi. and/or Jika Anda tertarik untuk mengeksekusi jenis kontak ini, silakan buka AWS Dukungan tiket melalui. AWS Support Center Console Kami akan membantu Anda menentukan konfigurasi dan profil misi yang tepat untuk kasus penggunaan Anda.

Konfigurasi ini merupakan node sumber dalam aliran data. Ini bertanggung jawab untuk mendigitalkan data frekuensi radio dan melakukan demodulasi dan decoding seperti yang ditentukan. Data yang dialirkan dari node ini akan mengikuti Format Demodulated/Decoded data/IP. Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi decode demod downlink antena menggunakan CloudFormation, the AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config AntennaDownlinkDemodDecodeConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`antennaDownlinkDemodDecodeConfig -> (structure)`)
- AntennaDownlinkDemodDecodeConfig Referensi API

Konfigurasi Uplink Antena

Anda dapat menggunakan konfigurasi uplink antena untuk mengonfigurasi antena untuk uplink selama kontak Anda. Mereka terdiri dari konfigurasi spektrum dengan frekuensi, polarisasi, dan target daya radiasi isotropik efektif (EIRP). Untuk informasi tentang cara mengonfigurasi kontak untuk uplink loopback, lihat. Antena Uplink Echo Config

Konfigurasi ini merupakan node tujuan dalam aliran data. Ini akan mengubah sinyal data frekuensi radio digital yang disediakan menjadi sinyal analog dan memancarkannya untuk diterima satelit Anda. Data yang dialirkan ke node ini diharapkan memenuhi Data/IP Format Sinyal. Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi uplink antena menggunakan CloudFormation, the AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config AntennaUplinkConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`antennaUplinkConfig -> (structure)`)
- AntennaUplinkConfig Referensi API

Antena Uplink Echo Config

Konfigurasi gema uplink memberi tahu antena cara menjalankan gema uplink. Gema uplink dapat digunakan untuk memvalidasi perintah yang dikirim ke pesawat ruang angkasa Anda, dan melakukan tugas-tugas lanjutan lainnya. Ini dicapai dengan merekam sinyal aktual yang ditransmisikan oleh AWS Ground Station antena (yaitu uplink). Ini menggemakan sinyal yang dikirim oleh antena kembali ke titik akhir aliran data Anda dan harus sesuai dengan sinyal yang ditransmisikan. Konfigurasi gema

uplink berisi ARN dari konfigurasi uplink. Antena menggunakan parameter dari konfigurasi uplink yang ditunjuk oleh ARN saat menjalankan gema uplink.

Konfigurasi ini merupakan node sumber dalam aliran data. Data yang dialirkan dari node ini akan memenuhi Data/IP Format Sinyal. Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi echo uplink menggunakan CloudFormation, the AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config UplinkEchoConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`uplinkEchoConfig -> (structure)`)
- UplinkEchoConfig Referensi API

Konfigurasi Titik Akhir Dataflow

> **Note**
>
> Konfigurasi titik akhir aliran data hanya digunakan untuk pengiriman data ke Amazon EC2 dan tidak digunakan untuk pengiriman data ke Amazon S3.

Anda dapat menggunakan konfigurasi titik akhir aliran data untuk menentukan titik akhir aliran data mana dalam grup titik akhir aliran data dari mana atau ke mana Anda ingin data mengalir selama kontak. Dua parameter konfigurasi titik akhir aliran data menentukan nama dan wilayah titik akhir aliran data. Saat memesan kontak, AWS Ground Station analisis profil misi yang Anda tentukan dan coba temukan grup titik akhir aliran data dalam AWS Wilayah yang berisi semua titik akhir aliran data yang ditentukan oleh konfigurasi titik akhir aliran data yang terdapat dalam profil misi Anda. Jika grup titik akhir aliran data yang sesuai ditemukan, status kontak akan menjadi DIJADWALKAN, jika tidak maka akan menjadi FAILED_TO_SCHEDULE. Untuk informasi lebih lanjut tentang kemungkinan status kontak, lihatAWS Ground Station status kontak.

`dataflowEndpointName`Properti konfigurasi titik akhir aliran data menentukan titik akhir aliran data mana dalam grup titik akhir aliran data ke mana atau dari mana data akan mengalir selama kontak.

`dataflowEndpointRegion`Properti menentukan wilayah mana titik akhir aliran data berada. Jika wilayah ditentukan dalam konfigurasi titik akhir aliran data Anda, AWS Ground Station cari titik akhir

aliran data di wilayah yang ditentukan. Jika tidak ada wilayah yang ditentukan, AWS Ground Station akan default ke wilayah stasiun bumi kontak. Kontak dianggap sebagai kontak pengiriman data lintas wilayah jika wilayah titik akhir aliran data Anda tidak sama dengan wilayah stasiun darat kontak. Lihat Bekerja dengan aliran data untuk informasi lebih lanjut tentang aliran data lintas wilayah.

Lihat Gunakan grup AWS Ground Station titik akhir Dataflow tips tentang bagaimana skema penamaan yang berbeda untuk aliran data Anda dapat bermanfaat bagi kasus penggunaan Anda.

Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi titik akhir aliran data menggunakan CloudFormation,, atau API AWS Command Line Interface. AWS Ground Station

- AWS::GroundStation::Config DataflowEndpointConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`dataflowEndpointConfig -> (structure)`)
- DataflowEndpointConfig Referensi API

Config Perekaman Amazon S3

> **Note**
>
> Konfigurasi perekaman Amazon S3 hanya digunakan untuk pengiriman data ke Amazon S3 dan tidak digunakan untuk pengiriman data ke Amazon EC2.

Konfigurasi ini merupakan node tujuan dalam aliran data. Node ini akan merangkum data yang masuk dari node sumber aliran data ke dalam data pcap. Untuk informasi lebih rinci tentang cara membuat aliran data dengan konfigurasi ini, lihat Bekerja dengan aliran data

Anda dapat menggunakan konfigurasi perekaman S3 untuk menentukan bucket Amazon S3 yang ingin dikirimkan data downlink bersama dengan konvensi penamaan yang digunakan. Berikut ini menentukan batasan dan rincian tentang parameter ini:

- Nama bucket Amazon S3 harus dimulai dengan. `aws-groundstation`
- Peran IAM harus memiliki kebijakan kepercayaan yang memungkinkan kepala `groundstation.amazonaws.com` layanan untuk mengambil peran tersebut. Lihat bagian

Contoh Kebijakan Kepercayaan di bawah ini untuk contoh. Selama pembuatan konfigurasi, id sumber daya konfigurasi tidak ada, kebijakan trust harus menggunakan tanda bintang (*) sebagai pengganti *your-config-id* dan dapat diperbarui setelah dibuat dengan id sumber daya konfigurasi.

Contoh Kebijakan Kepercayaan

Untuk informasi selengkapnya tentang cara memperbarui kebijakan kepercayaan peran, lihat Mengelola peran IAM di Panduan Pengguna IAM.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "999999999999"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:groundstation:us-
east-1:999999999999:config/s3-recording/your-config-id"
                }
            }
        }
    ]
}
```

- Peran IAM harus memiliki kebijakan IAM yang memungkinkan peran untuk melakukan `s3:GetBucketLocation` tindakan pada bucket dan `s3:PutObject` tindakan pada objek bucket. Jika bucket Amazon S3 memiliki kebijakan bucket, kebijakan bucket juga harus mengizinkan peran IAM untuk melakukan tindakan ini. Lihat bagian Kebijakan Peran Contoh di bawah ini untuk contoh.

Contoh Kebijakan Peran

Untuk informasi selengkapnya tentang cara memperbarui atau melampirkan kebijakan peran, lihat Mengelola kebijakan IAM di Panduan Pengguna IAM.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation"
      ],
      "Resource": [
        "arn:aws:s3:::your-bucket-name"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::your-bucket-name/*"
      ]
    }
  ]
}
```

- Awalan akan digunakan saat menamai objek data S3. Anda dapat menentukan kunci opsional untuk substitusi, nilai-nilai ini akan diganti dengan informasi yang sesuai dari detail kontak Anda. Misalnya, awalan `{satellite_id}/{year}/{month}/{day}` akan diganti dan akan dihasilkan dengan output seperti `fake_satellite_id/2021/01/10`

  Kunci opsional untuk substitusi: `{satellite_id}` | `{config-name}` | `{config-id}` | `{year}` | `{month}` | `{day}`

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi perekaman S3 menggunakan CloudFormation, API AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config Properti S3 RecordingConfig CloudFormation
- AWS CLI Referensi Config (lihat bagians3RecordingConfig -> (structure))
- Referensi RecordingConfig API S3

Konfigurasi Wastafel Telemetri

Anda dapat menggunakan konfigurasi sink telemetri untuk menentukan di mana Anda ingin data telemetri dikirimkan selama kontak satelit. Konfigurasi sink telemetri bersifat opsional dan ditambahkan ke profil misi Anda untuk menjadwalkan kontak berkemampuan telemetri. Berikut ini menentukan batasan dan rincian tentang parameter ini:

- Peran IAM harus memiliki kebijakan kepercayaan yang memungkinkan kepala `groundstation.amazonaws.com` layanan untuk mengambil peran tersebut. Lihat bagian Contoh Kebijakan Kepercayaan di bawah ini untuk contoh.

Contoh Kebijakan Kepercayaan

Untuk informasi selengkapnya tentang cara memperbarui kebijakan kepercayaan peran, lihat Mengelola peran IAM di Panduan Pengguna IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Peran IAM harus memiliki kebijakan IAM yang memungkinkan peran untuk melakukan`kinesis:DescribeStream`, `kinesis:PutRecord` dan `kinesis:PutRecords` tindakan di aliran. Lihat bagian Kebijakan Peran Contoh di bawah ini untuk contoh.

Contoh Kebijakan Peran

Untuk informasi selengkapnya tentang cara memperbarui atau melampirkan kebijakan peran, lihat Mengelola kebijakan IAM di Panduan Pengguna IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesis:DescribeStream",
        "kinesis:PutRecord",
        "kinesis:PutRecords"
      ],
      "Resource": "arn:aws:kinesis:us-east-2:999999999999:stream/your-stream-name"
    }
  ]
}
```

Saat Anda menyertakan konfigurasi sink telemetri di profil misi Anda, AWS Ground Station akan mengalirkan data telemetri ke akun Anda selama kontak. Untuk informasi selengkapnya tentang jenis telemetri, format data, dan pengaturan AWS sumber daya yang diperlukan, lihat. Bekerja dengan telemetri

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada konfigurasi sink telemetri menggunakan CloudFormation, the AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::Config TelemetrySinkConfig CloudFormation properti
- AWS CLI Referensi Config (lihat bagian`telemetrySinkConfig -> (structure)`)
- TelemetrySinkConfig Referensi API

Gunakan grup AWS Ground Station titik akhir Dataflow

Titik akhir Dataflow menentukan lokasi tempat Anda ingin data dialirkan secara sinkron ke atau dari selama kontak. Titik akhir aliran data selalu dibuat sebagai bagian dari grup endpoint aliran data. Dengan menyertakan beberapa titik akhir aliran data dalam grup, Anda menegaskan bahwa titik akhir yang ditentukan semuanya dapat digunakan bersama selama satu kontak. Misalnya, jika kontak perlu mengirim data ke tiga titik akhir aliran data terpisah, Anda harus memiliki tiga titik akhir dalam satu grup titik akhir aliran data yang cocok dengan konfigurasi titik akhir aliran data di profil misi Anda.

Versi grup titik akhir Dataflow

AWS Ground Station mendukung dua versi grup titik akhir aliran data:

- DataflowEndpointGroup- Implementasi asli yang mendukung uplink dan downlink menggunakan endpoint dataflow, dan downlink-only untuk titik akhir Agen AWS Ground Station
- DataflowEndpointGroupV2 - Versi terbaru yang mendukung aliran data uplink dan downlink untuk titik akhir AWS Ground Station Agen dengan kejelasan dan fungsionalitas yang ditingkatkan

Perbandingan grup titik akhir Dataflow

Fitur	DataflowEndpointGroup	DataflowEndpointGroupV2
Jenis titik akhir yang didukung	DataflowEndpoint, AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint, UplinkAwsGroundStationAgentEndpoint
Titik akhir yang mendukung uplink	DataflowEndpoint	UplinkAwsGroundStationAgentEndpoint
Endpoint yang mendukung downlink	DataflowEndpoint, AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint

DataflowEndpointGroupV2 dibuat untuk mendukung aliran data uplink dan untuk membuat bahasa di sekitar grup titik akhir aliran data lebih jelas. Kami merekomendasikan penggunaan UplinkAwsGroundStationAgentEndpointdan DownlinkAwsGroundStationAgentEndpointtitik akhir dengan DataflowEndpointGroupV2 untuk semua kasus penggunaan baru. DataflowEndpointGroup

tetap didukung untuk kompatibilitas mundur, tetapi DataflowEndpointGroup V2 menyediakan fungsionalitas yang ditingkatkan dan opsi konfigurasi yang lebih jelas.

> **Tip**
>
> Titik akhir aliran data diidentifikasi dengan nama yang Anda pilih saat menjalankan kontak. Nama-nama ini tidak harus unik di seluruh akun. Hal ini memungkinkan beberapa kontak di satelit dan antena yang berbeda untuk dieksekusi pada saat yang sama menggunakan profil misi yang sama. Ini dapat berguna jika Anda memiliki konstelasi satelit yang memiliki karakteristik operasi yang sama. Anda dapat menskalakan jumlah grup titik akhir aliran data agar sesuai dengan jumlah maksimum kontak simultan yang dibutuhkan oleh konstelasi satelit Anda.

Ketika satu atau beberapa sumber daya dalam grup titik akhir aliran data digunakan untuk kontak, seluruh grup dicadangkan selama durasi kontak tersebut. Anda dapat mengeksekusi beberapa kontak secara bersamaan, tetapi kontak tersebut harus dieksekusi pada grup endpoint aliran data yang berbeda.

> **Important**
>
> Grup titik akhir aliran data harus dalam `HEALTHY` keadaan untuk menjadwalkan kontak yang menggunakannya. Untuk informasi tentang cara memecahkan masalah grup titik akhir aliran data yang tidak berada dalam status, lihat. `HEALTHY` Memecahkan masalah DataflowEndpointGroups tidak dalam keadaan SEHAT

Lihat dokumentasi berikut untuk informasi selengkapnya tentang cara melakukan operasi pada grup endpoint aliran data yang menggunakan CloudFormation, API AWS Command Line Interface, atau API. AWS Ground Station

- AWS::GroundStation::DataflowEndpointGroup CloudFormation jenis sumber daya
- Referensi Dataflow Endpoint Group AWS CLI
- Referensi API Grup Titik Akhir Dataflow

Titik akhir aliran data

Anggota grup titik akhir aliran data adalah titik akhir aliran data. Jenis titik akhir yang didukung bergantung pada versi grup titik akhir aliran data yang Anda gunakan.

DataflowEndpointGroup titik akhir

DataflowEndpointGroup mendukung uplink dan downlink menggunakan endpoint dataflow, dan downlink-only untuk titik akhir Agen.AWS Ground Station Untuk kedua jenis titik akhir, Anda akan membuat konstruksi pendukung (misalnya alamat IP) sebelum membuat grup endpoint aliran data. Silakan lihat Bekerja dengan aliran data rekomendasi tentang jenis titik akhir aliran data mana yang akan digunakan dan cara mengatur konstruksi pendukung.

Bagian berikut menjelaskan kedua jenis endpoint yang didukung.

> **Important**
>
> Semua titik akhir aliran data dalam satu grup titik akhir aliran data harus dari jenis yang sama. Anda tidak dapat mencampur titik akhir AWS Ground Station Agen dengan titik akhir Dataflow dalam grup yang sama. Jika kasus penggunaan Anda memerlukan kedua jenis titik akhir, Anda harus membuat grup titik akhir aliran data terpisah untuk setiap jenis.
> Untuk DataflowEndpointGroup V2, Anda dapat mencampur UplinkAwsGroundStationAgentEndpointdan DownlinkAwsGroundStationAgentEndpointdalam grup yang sama.

AWS Ground Station Titik akhir agen

AWS Ground Station Agen Endpoint menggunakan AWS Ground Station Agen sebagai komponen perangkat lunak untuk mengakhiri koneksi. Untuk membangun Endpoint AWS Ground Station Agen, Anda hanya akan mengisi `AwsGroundStationAgentEndpoint` bidang. EndpointDetails Untuk informasi selengkapnya tentang AWS Ground Station Agen, lihat Panduan Pengguna AWS Ground Station Agen selengkapnya.

`AwsGroundStationAgentEndpoint`Terdiri dari yang berikut:

- `Name`- Nama titik akhir aliran data. Agar kontak dapat menggunakan titik akhir aliran data ini, nama ini harus cocok dengan nama yang digunakan dalam konfigurasi titik akhir aliran data Anda.
- `EgressAddress`- Alamat IP dan port yang digunakan untuk mengeluarkan data dari Agen.

- `IngressAddress`- Alamat IP dan port yang digunakan untuk memasukkan data ke Agen.

Titik akhir aliran data

Dataflow Endpoint menggunakan aplikasi jaringan sebagai komponen perangkat lunak untuk mengakhiri koneksi. Gunakan Dataflow Endpoint saat Anda ingin melakukan uplink Digital Signal Data, downlink kurang dari 50 MHz Data Sinyal Digital, atau Downlink Signal Data. Demodulated/ Decoded Untuk membangun Dataflow Endpoint, Anda akan mengisi dan bidang. `Endpoint Security Details` EndpointDetails

`Endpoint`Terdiri dari yang berikut:

- `Name`- Nama titik akhir aliran data. Agar kontak dapat menggunakan titik akhir aliran data ini, nama ini harus cocok dengan nama yang digunakan dalam konfigurasi titik akhir aliran data Anda.
- `Address`- Alamat IP dan port yang digunakan.

`SecurityDetails`Terdiri dari yang berikut:

- `roleArn`- Nama Sumber Daya Amazon (ARN) dari peran yang AWS Ground Station akan diasumsikan untuk membuat Antarmuka Jaringan Elastis (ENIs) di VPC Anda. Ini ENIs berfungsi sebagai titik masuk dan keluar dari data yang dialirkan selama kontak.
- `securityGroupIds`- Grup keamanan untuk dilampirkan ke antarmuka jaringan elastis.
- `subnetIds`- Daftar subnet di mana AWS Ground Station dapat menempatkan antarmuka jaringan elastis untuk mengirim aliran ke instans Anda. Jika beberapa subnet ditentukan, mereka harus dapat dirutekan satu sama lain. Jika subnet berada di Availability Zone (AZs) yang berbeda, Anda mungkin dikenakan biaya transfer data lintas-AZ.

Peran IAM yang diteruskan `roleArn` harus memiliki kebijakan kepercayaan yang memungkinkan kepala `groundstation.amazonaws.com` layanan untuk mengambil peran tersebut. Lihat bagian Contoh Kebijakan Kepercayaan di bawah ini untuk contoh. Selama pembuatan titik akhir, id sumber daya titik akhir tidak ada, jadi kebijakan kepercayaan harus menggunakan tanda bintang (`*`) sebagai pengganti. *`your-endpoint-id`* Ini dapat diperbarui setelah pembuatan untuk menggunakan id sumber daya titik akhir untuk cakupan kebijakan kepercayaan ke grup titik akhir aliran data tertentu.

Peran IAM harus memiliki kebijakan IAM yang memungkinkan AWS Ground Station untuk mengatur. ENIs Lihat bagian Kebijakan Peran Contoh di bawah ini untuk contoh.

Contoh Kebijakan Kepercayaan

Untuk informasi selengkapnya tentang cara memperbarui kebijakan kepercayaan peran, lihat Mengelola peran IAM di Panduan Pengguna IAM.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "999999999999"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:groundstation:us-
east-1:999999999999:dataflow-endpoint-group/your-endpoint-id"
                }
            }
        }
    ]
}
```

Contoh Kebijakan Peran

Untuk informasi selengkapnya tentang cara memperbarui atau melampirkan kebijakan peran, lihat Mengelola kebijakan IAM di Panduan Pengguna IAM.

JSON

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

```
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
        "ec2:CreateNetworkInterfacePermission",
        "ec2:DeleteNetworkInterfacePermission",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs",
        "ec2:DescribeSecurityGroups"
      ],
      "Resource": "*"
    }
  ]
}
```

DataflowEndpointGroupTitik akhir V2

DataflowEndpointGroupV2 memperkenalkan tipe titik akhir khusus yang menyediakan konfigurasi yang lebih jelas dan fungsionalitas yang ditingkatkan:

- UplinkAwsGroundStationAgentEndpoint- Dioptimalkan untuk aliran data uplink
- DownlinkAwsGroundStationAgentEndpoint- Dioptimalkan untuk aliran data downlink

Titik akhir khusus ini menggantikan generik AwsGroundStationAgentEndpointdengan konfigurasi khusus arah yang membuatnya lebih mudah untuk mengatur dan mengelola aliran data Anda.

Titik akhir Agen Uplink AWS Ground Station

UplinkAwsGroundStationAgentEndpointIni dirancang khusus untuk aliran data uplink dan menyediakan opsi konfigurasi yang lebih jelas. Gunakan tipe titik akhir ini saat Anda perlu memberikan data untuk di-uplink AWS Ground Station ke satelit Anda.

`UplinkAwsGroundStationAgentEndpoint`Terdiri dari yang berikut:

- `Name`- Nama titik akhir aliran data. Agar kontak dapat menggunakan titik akhir aliran data ini, nama ini harus cocok dengan nama yang digunakan dalam konfigurasi titik akhir aliran data Anda.
- `IngressAddressAndPort`- IP tunggal dan alamat port untuk input data ke agen
- `AgentIpAndPortAddress`- Jangkauan port untuk komunikasi agen

Titik akhir AWS Ground Station Agen Downlink

DownlinkAwsGroundStationAgentEndpointIni dioptimalkan untuk aliran data downlink, termasuk downlink narrowband, demodulasi/decode pita lebar, dan skenario gema uplink.

`DownlinkAwsGroundStationAgentEndpoint`Terdiri dari yang berikut:

- `Name`- Nama titik akhir aliran data. Agar kontak dapat menggunakan titik akhir aliran data ini, nama ini harus cocok dengan nama yang digunakan dalam konfigurasi titik akhir aliran data Anda.
- `EgressAddressAndPort`- IP tunggal dan alamat port untuk output data dari agen
- `AgentIpAndPortAddress`- Jangkauan port untuk komunikasi agen

Membuat grup titik akhir aliran data

Anda dapat membuat grup titik akhir aliran data menggunakan salah satu versi:

CreateDataflowEndpointGroup

Gunakan CreateDataflowEndpointGroupuntuk kompatibilitas mundur atau ketika Anda perlu menggunakan generik AwsGroundStationAgentEndpointatau DataflowEndpointtipe.

CreateDataflowEndpointGroupV2

Gunakan CreateDataflowEndpointGroupV2 untuk implementasi baru guna memanfaatkan tipe endpoint khusus yang mendukung aliran data uplink dan downlink. API ini hanya mendukung UplinkAwsGroundStationAgentEndpointdan DownlinkAwsGroundStationAgentEndpoint.

Pertimbangan migrasi

Jika saat ini Anda menggunakan DataflowEndpointGroup, Anda dapat terus menggunakan konfigurasi yang ada tanpa perubahan. AWS Ground Station mempertahankan kompatibilitas mundur penuh.

Jika Anda ingin bermigrasi untuk menggunakan DataflowEndpointGroup V2 baru, dan saat ini menggunakan Aplikasi DataflowEndpointdengan Dataflow Endpoint untuk menerima data Anda, Anda harus bermigrasi untuk menggunakan Agen sebagai gantinya. AWS Ground Station Jika Anda sudah menggunakan AWS Ground Station Agen untuk downlink, Anda dapat menggunakan instance agen yang sama untuk uplink juga - tidak diperlukan instance agen tambahan.

Untuk bermigrasi ke DataflowEndpointGroup V2:

1. Jika bermigrasi dari DataflowEndpoint, siapkan AWS Ground Station Agen mengikuti Panduan Pengguna AWS Ground Station Agen
2. Identifikasi arah aliran data Anda dan buat tipe titik akhir yang sesuai (atau) UplinkAwsGroundStationAgentEndpointDownlinkAwsGroundStationAgentEndpoint
3. Buat DataflowEndpointGroupV2 yang mereferensikan titik akhir tersebut
4. Buat konfigurasi titik akhir aliran data baru yang mereferensikan V2 baru berdasarkan nama DataflowEndpointGroup
5. Buat profil misi baru yang mereferensikan konfigurasi titik akhir aliran data sebagai tepi aliran data
6. Gunakan profil misi baru untuk menjadwalkan kontak
7. Uji konfigurasi Anda sebelum menerapkan ke produksi

Untuk informasi selengkapnya tentang alur kerja lengkap, lihat Memahami komponen AWS Ground Station Inti danBuat konfigurasi.

Gunakan AWS Ground Station Agen

AWS Ground Station Agen memungkinkan Anda menerima aliran data Wideband Digital Intermediate Frequency (DiGIF) sinkron (downlink) selama kontak AWS Ground Station.

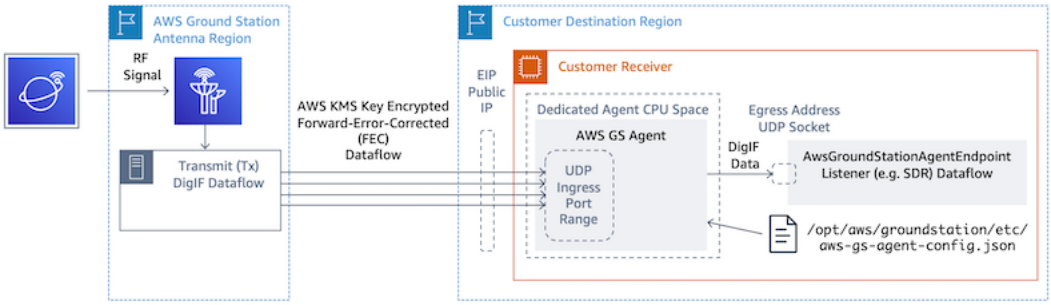
Cara kerjanya

Anda dapat memilih dua opsi untuk pengiriman data:

1. Pengiriman data ke EC2 instance - Pengiriman data ke EC2 instance yang Anda miliki. Anda mengelola AWS Ground Station Agen. Opsi ini mungkin paling cocok untuk Anda jika Anda membutuhkan pemrosesan data mendekati waktu nyata. Lihat Bekerja dengan aliran data bagian untuk informasi tentang pengiriman EC2 data.
2. Pengiriman data ke bucket S3 - Pengiriman data ke bucket AWS S3 Anda dikelola sepenuhnya oleh. AWS Ground Station Lihat Memulai panduan untuk informasi tentang pengiriman data S3.

Kedua mode pengiriman data mengharuskan Anda membuat satu set sumber daya AWS. Penggunaan CloudFormation untuk membuat sumber daya AWS Anda sangat disarankan untuk memastikan keandalan, akurasi, dan dukungan. Setiap kontak hanya dapat mengirimkan data ke EC2 atau S3 tetapi tidak ke keduanya secara bersamaan.

Diagram berikut menunjukkan aliran data DiGIF dari Wilayah AWS Ground Station Antena ke EC2 instans Anda dengan Software-Defined Radio (SDR) atau pendengar serupa.



Informasi tambahan

Untuk informasi lebih lanjut, silakan lihat Panduan Pengguna AWS Ground Station Agen lengkap.

Memulai

Sebelum Anda mulai, Anda harus membiasakan diri dengan konsep dasar di AWS Ground Station. Untuk informasi selengkapnya, lihat Bagaimana cara AWS Ground Station kerja.

Mendaftar untuk Akun AWS

Untuk memulai AWS, Anda membutuhkan Akun AWS. Untuk informasi tentang membuat Akun AWS, lihat Memulai dengan Akun AWS di Panduan AWS Account Management Referensi.

Tambahkan AWS Ground Station izin untuk Anda AWS akun

Untuk menggunakan AWS Ground Station tanpa memerlukan pengguna administratif, Anda perlu membuat kebijakan baru dan melampirkannya ke AWS akun Anda.

1. Masuk ke Konsol Manajemen AWS dan buka konsol IAM.
2. Membuat kebijakan baru. Gunakan langkah-langkah berikut:

    a. Di panel navigasi, pilih Kebijakan, lalu pilih Buat Kebijakan.

    b. Di tab JSON, edit JSON dengan salah satu nilai berikut. Gunakan JSON yang paling sesuai untuk aplikasi Anda.

    - Untuk hak administratif Ground Station, atur Action ke groundstation: * sebagai berikut:

      JSON

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

- Untuk Read-only hak istimewa, setel Action ke GroundStation:get*, GroundStation:list*, dan groundStation:describe* sebagai berikut:

  JSON

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

- Untuk keamanan tambahan melalui otentikasi multifaktor, atur Action ke groundstation: *, dan Condition/Boolke aws: sebagai berikut: MultiFactorAuthPresent:true

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "groundstation:*",
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "aws:MultiFactorAuthPresent": true
                }
            }
        }
    ]
```

```
}
```

3. Di konsol IAM, lampirkan kebijakan yang Anda buat ke pengguna yang diinginkan.

Untuk informasi selengkapnya tentang pengguna IAM dan melampirkan kebijakan, lihat Panduan Pengguna IAM.

Satelit onboard

Memasukkan satelit ke dalam AWS Ground Station adalah proses multistep yang melibatkan pengumpulan data, validasi teknis, lisensi spektrum, dengan integrasi dan pengujian. Ada juga perjanjian non-pengungkapan (NDAs) yang diperlukan.

Ikhtisar proses orientasi pelanggan

Orientasi satelit adalah proses manual yang dapat ditemukan di bagian Satelit dan Sumber Daya di halaman konsol. AWS Ground Station Berikut ini menjelaskan keseluruhan proses.

1. Tinjau AWS Ground Station Lokasi bagian untuk menentukan apakah satelit Anda memenuhi karakteristik frekuensi geografis dan radio.
2. Untuk memulai orientasi satelit Anda AWS Ground Station, kirimkan Kuesioner Orientasi Satelit di bagian Satelit dan Sumber Daya di halaman konsol. AWS Ground Station Harap sertakan ringkasan singkat tentang misi dan kebutuhan satelit Anda, termasuk nama organisasi Anda, frekuensi yang diperlukan, kapan satelit akan atau diluncurkan, jenis orbit satelit, dan jika Anda berencana untuk menggunakannya. Gunakan AWS Ground Station fitur kembar digital
3. Setelah permintaan Anda ditinjau dan disetujui, AWS Ground Station akan mengajukan permohonan lisensi peraturan di lokasi tertentu yang Anda rencanakan untuk digunakan. Durasi langkah ini akan bervariasi tergantung pada lokasi dan peraturan yang ada.
4. Setelah persetujuan ini diperoleh, satelit Anda akan terlihat untuk Anda gunakan. AWS Ground Station akan mengirimi Anda pemberitahuan tentang pembaruan yang berhasil.

(Opsional) Penamaan satelit

Setelah onboarding, Anda mungkin ingin menambahkan nama ke catatan satelit Anda agar lebih mudah mengenalinya. AWS Ground Station Konsol memiliki kemampuan untuk menampilkan nama yang ditentukan pengguna untuk satelit bersama dengan ID Norad saat menggunakan halaman

Kontak. Menampilkan nama satelit membuatnya lebih mudah untuk memilih satelit yang benar saat menjadwalkan. Untuk melakukan ini, tag dapat digunakan.

Menandai AWS Ground Station Satellites dapat dilakukan melalui API tag-resource dengan AWS CLI atau salah satu AWS. SDKs Panduan ini akan mencakup penggunaan AWS Ground Station CLI untuk menandai satelit siaran publik Aqua (Norad ID 27424) di. `us-west-2`

AWS Ground Station CLI

Hal ini AWS CLI dapat digunakan untuk berinteraksi dengan AWS Ground Station Sebelum menggunakan AWS CLI untuk menandai satelit Anda, AWS CLI prasyarat berikut harus dipenuhi:

- Pastikan AWS CLI sudah terpasang. Untuk informasi tentang penginstalan AWS CLI, lihat Menginstal AWS CLI versi 2.
- Pastikan itu AWS CLI dikonfigurasi. Untuk informasi tentang mengonfigurasi AWS CLI, lihat Mengonfigurasi AWS CLI versi 2.
- Simpan pengaturan konfigurasi dan kredensil yang sering Anda gunakan dalam file yang dikelola oleh file. AWS CLI Anda memerlukan pengaturan dan kredensil ini untuk memesan dan mengelola AWS Ground Station kontak Anda. AWS CLI Untuk informasi selengkapnya tentang menyimpan konfigurasi dan setelan kredensialnya, lihat Pengaturan konfigurasi dan file kredensi.

Setelah AWS CLI dikonfigurasi dan siap digunakan, tinjau halaman AWS Ground Station CLI Command Reference untuk membiasakan diri dengan perintah yang tersedia. Ikuti struktur AWS CLI perintah saat menggunakan layanan ini dan awali perintah Anda `groundstation` untuk menentukan AWS Ground Station sebagai layanan yang ingin Anda gunakan. Untuk informasi selengkapnya tentang struktur AWS CLI perintah, lihat Struktur Perintah di halaman AWS CLI. Contoh struktur perintah disediakan di bawah ini.

```
aws groundstation <command> <subcommand> [options and parameters]
```

Nama Satelit

Pertama, Anda perlu mendapatkan ARN untuk satelit yang ingin Anda tag. Ini dapat dilakukan melalui API daftar-satelit di AWS CLI:

```
aws groundstation list-satellites --region us-west-2
```

Menjalankan perintah CLI di atas akan mengembalikan output yang mirip dengan ini:

```
{
    "satellites": [
        {
            "groundStations": [
                "Ohio 1",
                "Oregon 1"
            ],
            "noradSatelliteID": 27424,
            "satelliteArn":
 "arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555",
            "satelliteId": "11111111-2222-3333-4444-555555555555"
        }
    ]
}
```

Temukan satelit yang ingin Anda tandai dan catat`satelliteArn`. Satu peringatan penting untuk penandaan adalah bahwa API tag-resource memerlukan ARN regional, dan ARN yang dikembalikan oleh daftar-satelit bersifat global. Untuk langkah selanjutnya, Anda harus menambah ARN dengan wilayah tempat Anda ingin melihat tag (kemungkinan wilayah yang Anda jadwalkan). Untuk contoh ini, kami menggunakan`us-west-2`. Dengan perubahan ini, ARN akan berubah dari:

```
arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555
```

ke:

```
arn:aws:groundstation:us-
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555
```

Untuk menunjukkan nama satelit di konsol, satelit harus memiliki tag `"Name"` dengan kunci. Selain itu, karena kita menggunakan AWS CLI, tanda kutip harus diloloskan dengan garis miring terbalik. Tag akan terlihat seperti:

```
{\"Name\":\"AQUA\"}
```

Selanjutnya, Anda akan memanggil API tag-resource untuk menandai satelit. Hal ini dapat dilakukan dengan AWS CLI sejenisnya:

```
aws groundstation tag-resource --region us-west-2 --resource-arn
 arn:aws:groundstation:us-
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tags
 '{"Name":"AQUA"}'
```

Setelah melakukan ini, Anda akan dapat melihat nama yang Anda tetapkan untuk satelit di AWS Ground Station konsol.

Ubah Nama Untuk Satelit

Jika Anda ingin mengubah nama untuk satelit, Anda cukup memanggil tag-resource dengan ARN satelit lagi dengan "Name" kunci yang sama, tetapi dengan nilai yang berbeda dalam tag. Ini akan memperbarui tag yang ada dan menampilkan nama baru di konsol. Contoh panggilan untuk ini terlihat seperti:

```
aws groundstation tag-resource --region us-west-2 --resource-arn
 arn:aws:groundstation:us-
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tags
 '{"Name":"NewName"}'
```

Hapus Nama Untuk Satelit

Nama yang ditetapkan untuk satelit dapat dihapus dengan API untag-resource. API ini membutuhkan ARN satelit dengan wilayah tempat tag berada, dan daftar kunci tag. Untuk nama, kunci tag adalah"Name". Contoh panggilan ke API ini menggunakan AWS CLI terlihat seperti:

```
aws groundstation untag-resource --region us-west-2 --resource-arn
 arn:aws:groundstation:us-
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tag-keys Name
```

Satelit siaran publik

Selain orientasi satelit Anda sendiri, Anda dapat meminta untuk onboard dengan satelit siaran publik yang didukung yang menyediakan jalur komunikasi downlink yang dapat diakses publik. Ini memungkinkan Anda untuk menggunakan AWS Ground Station untuk downlink data dari satelit ini.

> **Note**
>
> Anda tidak akan dapat melakukan uplink ke satelit-satelit ini. Anda hanya akan dapat menggunakan jalur komunikasi downlink yang dapat diakses publik.

AWS Ground Station mendukung orientasi satelit berikut untuk menurunkan data siaran langsung:

- Aqua
- SNPP
- JPSS-1/NOAA-20
- Terra

Setelah onboard, satelit ini dapat diakses untuk segera digunakan. AWS Ground Station memelihara sejumlah CloudFormation template yang telah dikonfigurasi untuk membuat memulai dengan layanan lebih mudah. Lihat Contoh konfigurasi profil misi contoh bagaimana AWS Ground Station bisa digunakan.

Untuk informasi lebih lanjut tentang satelit ini dan jenis data yang mereka kirimkan, lihat Aqua, JPSS-1/NOAA-20 dan SNPP, dan Terra.

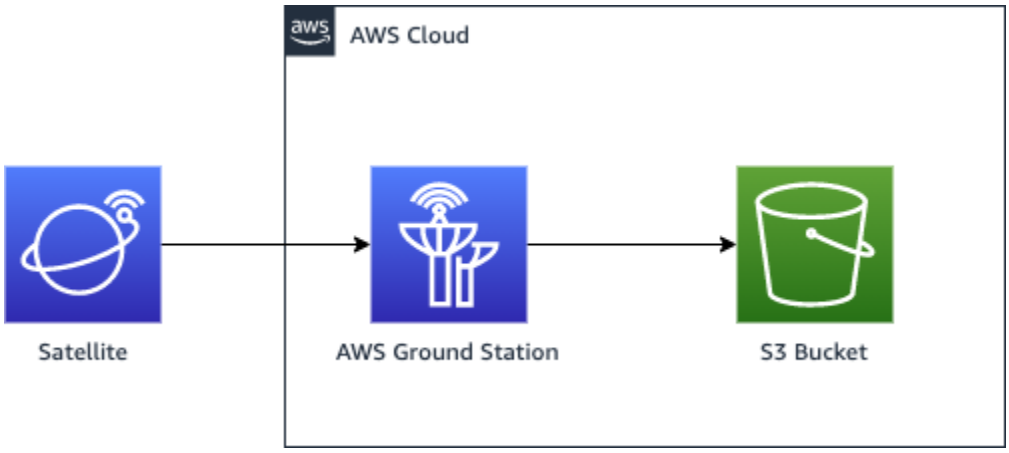
Rencanakan jalur komunikasi aliran data Anda

Anda memiliki pilihan antara komunikasi sinkron dan asinkron untuk setiap jalur komunikasi di satelit Anda. Tergantung pada satelit Anda dan kasus penggunaan Anda, Anda mungkin memerlukan satu atau kedua jenis. Jalur komunikasi sinkron memungkinkan uplink mendekati waktu nyata serta operasi downlink narrowband dan wideband. Jalur komunikasi asinkron hanya mendukung operasi downlink narrowband dan wideband.

Pengiriman data asinkron

Dengan pengiriman data ke Amazon S3, data kontak Anda dikirimkan secara asinkron ke bucket Amazon S3 di akun Anda. Data kontak Anda dikirimkan sebagai file packet capture (pcap) untuk memungkinkan pemutaran ulang data kontak ke Software Defined Radio (SDR) atau untuk mengekstrak data payload dari file pcap untuk diproses. File PCAP dikirim ke bucket Amazon S3 Anda setiap 30 detik karena data kontak diterima oleh perangkat keras antena untuk memungkinkan

pemrosesan data kontak selama kontak jika diinginkan. Setelah diterima, Anda dapat memproses data menggunakan perangkat lunak pasca-pemrosesan Anda sendiri atau menggunakan layanan AWS lainnya seperti Amazon SageMaker AI atau Amazon Rekognition. Pengiriman data ke Amazon S3 hanya tersedia untuk downlink data dari satelit Anda; tidak mungkin untuk menautkan data ke satelit Anda dari Amazon S3.



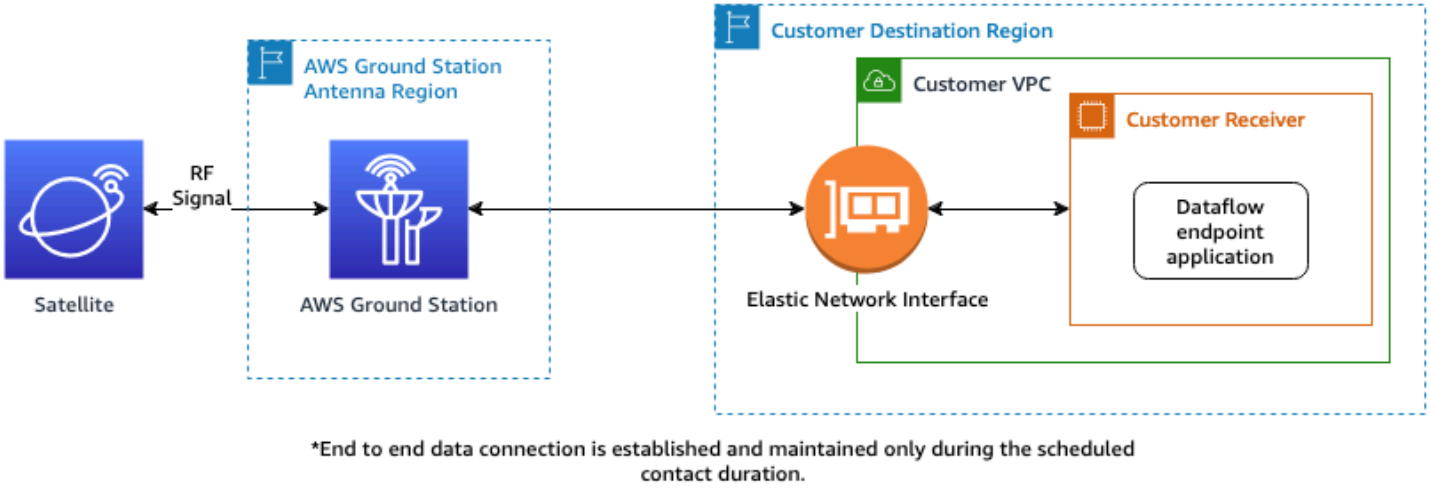
Untuk memanfaatkan jalur ini, Anda akan menggunakan kebutuhan untuk membuat bucket Amazon S3 AWS Ground Station untuk mengirimkan data ke dalamnya. Pada langkah berikutnya, Anda juga harus membuat S3 Recording Config di langkah berikutnya. Silakan referensi Config Perekaman Amazon S3 untuk pembatasan penamaan bucket dan cara menentukan konvensi penamaan yang digunakan untuk file Anda.

Pengiriman data sinkron

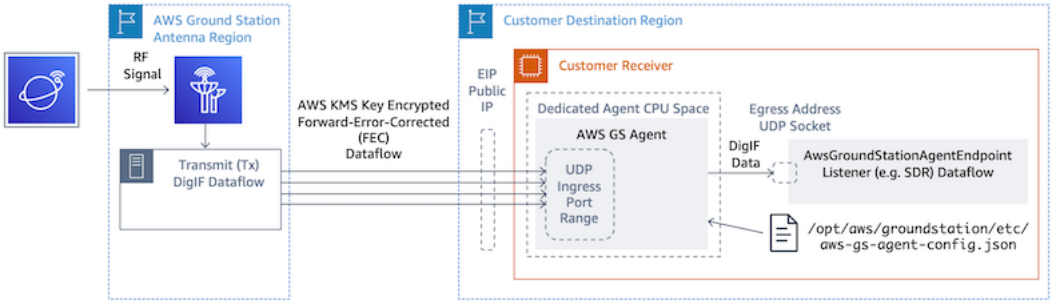
Dengan pengiriman data ke Amazon EC2, data kontak Anda dialirkan ke dan dari instans Amazon EC2 Anda. Anda dapat memproses data secara real-time di instans Amazon EC2 atau meneruskan data untuk pasca-pemrosesan.

Untuk memanfaatkan jalur sinkron, Anda akan menggunakan kebutuhan untuk mengatur dan mengonfigurasi instans Amazon EC2 Anda dan membuat satu atau beberapa Grup Titik Akhir Dataflow. Untuk mengonfigurasi referensi instans Amazon EC2, gunakan. Siapkan dan konfigurasikan Amazon EC2 Untuk membuat Grup Titik Akhir Dataflow Anda, silakan rujuk. Gunakan grup AWS Ground Station titik akhir Dataflow

Berikut ini menunjukkan jalur komunikasi jika Anda menggunakan konfigurasi titik akhir aliran data.



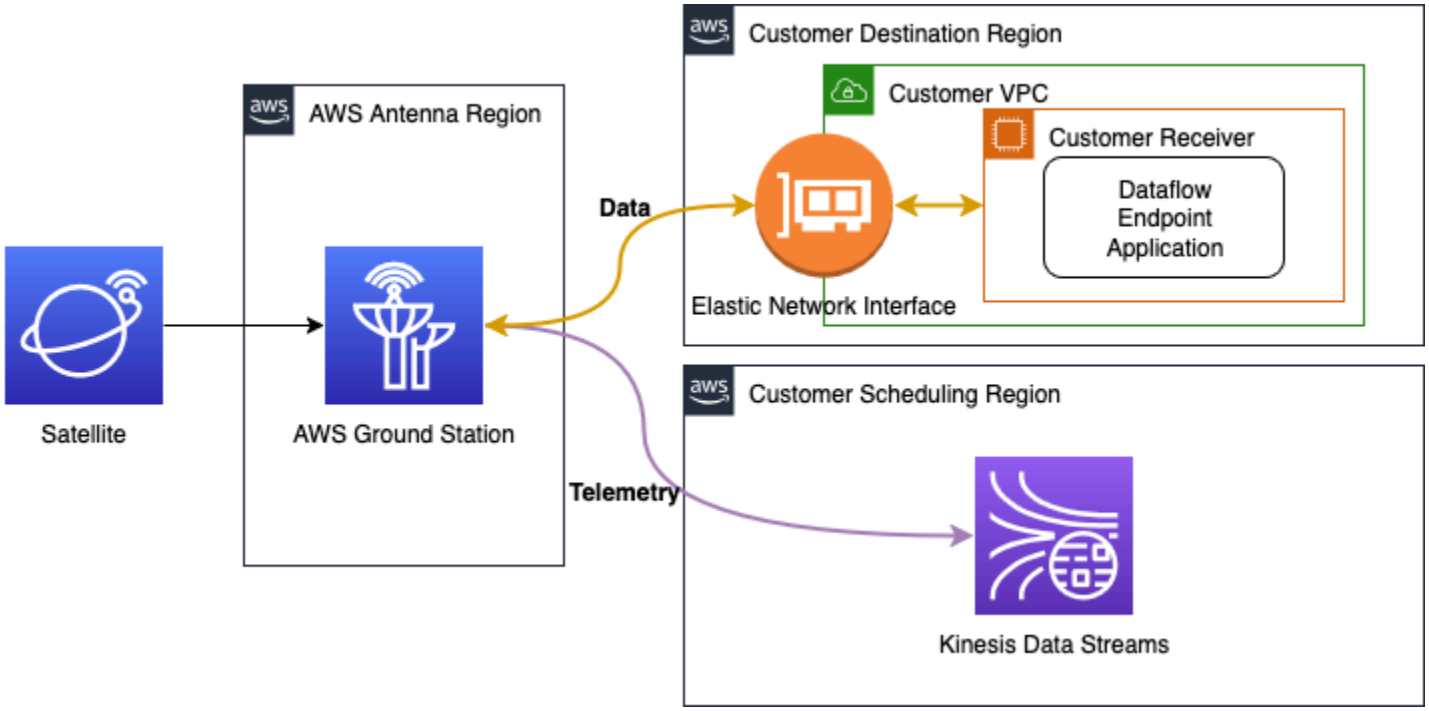
Berikut ini menunjukkan jalur komunikasi jika Anda menggunakan konfigurasi AWS Ground Station Agen.



Rencanakan telemetri Anda

AWS Ground Station telemetri adalah fitur opsional yang mengalirkan metrik dari AWS Ground Station antena ke akun Anda AWS selama kontak satelit. Ini memungkinkan Anda untuk memantau kinerja kontak dalam waktu dekat dan membangun solusi pemantauan khusus.

Dengan AWS Ground Station telemetri, metrik dari AWS Ground Station antena dialirkan langsung ke akun Anda. Data telemetri mulai streaming saat kontak dimulai dan berlanjut sepanjang durasi kontak. Data telemetri dikirim ke akun Anda dalam waktu dekat saat diambil sampelnya dari perangkat keras antena. Setelah diterima, Anda dapat memproses data menggunakan perangkat lunak pasca-pemrosesan Anda sendiri atau menggunakan layanan AWS lainnya seperti Amazon Data AWS Lambda Firehose atau.



Pada langkah berikutnya, Anda akan membuat konfigurasi yang diperlukan untuk profil misi Anda. Jika Anda ingin mengaktifkan telemetri, Anda akan membuat Telemetry Sink Config selain konfigurasi pelacakan dan konfigurasi aliran data Anda. Untuk petunjuk penyiapan terperinci, lihatSiapkan telemetri.

Untuk informasi lebih lanjut tentang TelemetrySinkConfig, lihatKonfigurasi Wastafel Telemetri.

Buat konfigurasi

Dengan langkah ini Anda telah mengidentifikasi satelit, jalur komunikasi, dan sumber daya IAM, Amazon EC2, dan Amazon S3 sesuai kebutuhan. Pada langkah ini Anda akan membuat AWS Ground Station konfigurasi yang menyimpan parameter masing-masing.

Konfigurasi pengiriman data

Konfigurasi pertama yang dibuat berhubungan dengan di mana dan bagaimana Anda ingin data dikirimkan. Menggunakan informasi dari langkah sebelumnya Anda akan membangun banyak jenis konfigurasi berikut.

- Config Perekaman Amazon S3- Kirimkan data ke bucket Amazon S3 Anda.
- Konfigurasi Titik Akhir Dataflow- Kirimkan data ke instans Amazon EC2 Anda.

Konfigurasi telemetri (opsional)

Jika Anda ingin menerima telemetri mendekati waktu nyata selama kontak Anda, Anda dapat membuat. TelemetrySinkConfig Konfigurasi ini opsional dan menentukan di mana AWS Ground Station akan mengirimkan data telemetri.

- Konfigurasi Wastafel Telemetri- Kirimkan data telemetri ke akun Anda.

Untuk petunjuk penyiapan terperinci, lihatSiapkan telemetri.

Konfigurasi satelit

Konfigurasi satelit menghubungkan bagaimana AWS Ground Station dapat berkomunikasi dengan satelit Anda. Anda akan mereferensikan informasi yang Anda kumpulkanSatelit onboard.

- Melacak Config- Menetapkan preferensi untuk bagaimana kendaraan Anda dilacak secara fisik selama kontak. Ini diperlukan untuk konstruksi profil misi.
- Konfigurasi Downlink Antena- Mengirimkan data frekuensi radio digital.
- Antena Downlink Demod Decode Config- Memberikan data frekuensi radio yang didemodulasi dan diterjemahkan.
- Konfigurasi Uplink Antena- Uplink data ke satelit Anda.
- Antena Uplink Echo Config- Kirimkan gema data sinyal uplink Anda.

Buat profil misi

Dengan konfigurasi yang dibangun pada langkah sebelumnya, Anda telah mengidentifikasi cara melacak satelit Anda, cara yang mungkin untuk berkomunikasi dengan satelit Anda, dan cara mengaktifkan telemetri mendekati waktu nyata selama eksekusi kontak. Pada langkah ini Anda akan membangun satu atau lebih profil misi. Profil misi mewakili agregasi konfigurasi yang mungkin ke dalam perilaku yang diharapkan yang kemudian dapat dijadwalkan dan dioperasikan.

Untuk parameter terbaru, silakan referensi jenis AWS::GroundStation::MissionProfile CloudFormation sumber daya

1. Beri nama profil misi Anda. Ini memungkinkan Anda untuk dengan cepat memahami penggunaannya dalam sistem Anda. Misalnya, Anda mungkin memiliki satellite-wideband-

narrowband-nominal-operasi dan satellite-narrowband-emergency-operationsjika Anda memiliki pembawa pita sempit terpisah untuk operasi darurat.

2. Setel konfigurasi pelacakan Anda.
3. Tetapkan durasi kontak minimum Anda yang layak. Ini memungkinkan Anda untuk memfilter kontak potensial untuk memenuhi kebutuhan misi Anda.
4. Atur streamsKmsKeydan streamsKmsRoleyang digunakan untuk mengenkripsi data Anda selama transit. Ini digunakan untuk semua aliran data AWS Ground Station Agen.
5. Tetapkan aliran data Anda. Buat aliran data Anda agar sesuai dengan sinyal operator Anda menggunakan konfigurasi yang Anda buat di langkah sebelumnya.
6. [Opsional] Atur detik durasi kontak pra-pass dan pasca-pass Anda. Ini digunakan untuk memancarkan peristiwa per-kontak sebelum dan sesudah kontak, masing-masing. Untuk informasi selengkapnya, lihat Otomatisasi AWS Ground Station dengan Acara.
7. [Opsional] Setel telemetrySinkConfigArn Anda untuk mengaktifkan telemetri selama kontak. Ini memungkinkan Anda menerima telemetri hampir real-time langsung di akun Anda untuk pemantauan dan analisis. Untuk informasi selengkapnya, lihat Bekerja dengan telemetri.
8. [Opsional] Anda dapat mengaitkan Tag ke profil misi Anda. Ini dapat digunakan untuk membantu membedakan profil misi Anda secara terprogram.

Anda dapat mereferensikanContoh konfigurasi profil misi, untuk melihat hanya beberapa konfigurasi potensial.

Pahami langkah selanjutnya

Sekarang setelah Anda memiliki satelit onboard dan profil misi yang valid, Anda siap untuk menjadwalkan kontak dan berkomunikasi dengan satelit Anda. AWS Ground Station

Anda dapat menjadwalkan kontak dengan salah satu cara berikut:

- AWS Ground Station Konsol.
- Perintah AWS kontak cadangan CLI.
- AWS SDK. ReserveContact API.

Untuk informasi tentang bagaimana AWS Ground Station melacak lintasan satelit Anda dan bagaimana informasi itu digunakan, silakan referensi. Memahami caranya AWS Ground Station menggunakan ephemerides

AWS Ground Station memelihara sejumlah CloudFormation template yang telah dikonfigurasi untuk membuat memulai dengan layanan lebih mudah. Lihat Contoh konfigurasi profil misi contoh bagaimana AWS Ground Station bisa digunakan.

Memproses data frekuensi menengah digital, atau data yang didemodulasi dan diterjemahkan yang diberikan kepada Anda dari AWS Ground Station akan tergantung pada kasus penggunaan spesifik Anda. Posting blog berikut dapat membantu Anda memahami beberapa opsi yang tersedia untuk Anda:

- Pengamatan Bumi otomatis menggunakan pengiriman data AWS Ground Station Amazon S3 (dan itu terkait GitHub repositori awslabs/) aws-groundstation-eos-pipeline
- Virtualisasi segmen darat satelit dengan AWS
- Pengamatan bumi menggunakan AWS Ground Station: Cara memandu
- Membangun arsitektur downlink data satelit throughput tinggi dengan AWS Ground Station WideBand DiGIF dan Amphinicy Blink SDR (dan repositori terkait aws-samples/) GitHub aws-groundstation-wbdigif-snpp

AWS Ground Station Lokasi

AWS Ground Station menyediakan jaringan global stasiun bumi yang dekat dengan jaringan global wilayah infrastruktur AWS kami. Anda dapat mengonfigurasi penggunaan lokasi ini dari Wilayah AWS yang didukung. Ini termasuk Wilayah AWS tempat data dikirimkan.

Menemukan AWS wilayah untuk lokasi stasiun bumi

Jaringan AWS Ground Station global mencakup lokasi stasiun bumi yang tidak secara fisik terletak di Wilayah AWS tempat mereka terhubung. Daftar stasiun bumi yang dapat Anda akses dapat diambil melalui respons AWS SDK ListGroundStation. Per 16 Juni 2026,AWS Ground Station tidak lagi mendukung sumber daya antena di Seoul. Daftar lengkap lokasi stasiun bumi disajikan di bawah ini. Silakan merujuk ke panduan orientasi untuk menambah atau memodifikasi persetujuan situs untuk satelit Anda.

Nama Ground Station	Lokasi Ground Station	Nama Wilayah AWS	Kode Wilayah AWS	Catatan
Alaska 1	Alaska, Amerika Serikat	AS Barat (Oregon)	us-west-2	Tidak secara fisik terletak

Nama Ground Station	Lokasi Ground Station	Nama Wilayah AWS	Kode Wilayah AWS	Catatan
				di suatu AWS wilayah
Bahrain 1	Bahrain	Timur Tengah (Bahrain)	me-south-1	
Kota Tanjung 1	Cape Town, Afrika Selatan	Africa (Cape Town)	af-south-1	
Dubbo 1	Dubbo, Australia	Asia Pasifik (Sydney)	ap-southeast-2	Tidak secara fisik terletak di suatu AWS wilayah
Hawaii 1	Hawaii, Amerika Serikat	AS Barat (Oregon)	us-west-2	Tidak secara fisik terletak di suatu AWS wilayah
Irlandia 1	Ireland	Eropa (Irlandia)	eu-west-1	
Ohio 1	Ohio, Amerika Serikat	AS Timur (Ohio)	us-east-2	
Oregon 1	Oregon, Amerika Serikat	AS Barat (Oregon)	us-west-2	
Punta Arena 1	Punta Arenas, Cile	Amerika Selatan (Sao Paulo)	sa-east-1	Tidak secara fisik terletak di suatu AWS wilayah
Singapura 1	Singapura	Asia Pasifik (Singapura)	ap-southeast-1	

Nama Ground Station	Lokasi Ground Station	Nama Wilayah AWS	Kode Wilayah AWS	Catatan
Stockholm 1	Stockholm, Swedia	Eropa (Stockholm)	eu-north-1	

AWS Ground Station wilayah AWS yang didukung

Anda dapat mengirimkan data dan mengonfigurasi Kontak Anda melalui AWS SDK atau AWS Ground Station konsol dari Wilayah AWS yang didukung. Anda dapat melihat wilayah yang didukung dan titik akhir terkait di AWS Ground Station titik akhir dan kuota.

Ketersediaan kembar digital

Gunakan AWS Ground Station fitur kembar digitaltersedia di semua Wilayah AWSAWS Ground Station jika tersedia. Stasiun ground kembar digital adalah salinan persis dari stasiun bumi produksi dengan awalan modifikasi ke Ground Station Nama “Digital Twin”. Misalnya, “Digital Twin Ohio 1" adalah stasiun darat kembar digital yang merupakan salinan persis dari stasiun bumi produksi “Ohio 1".

Antena Khusus

Selain lokasi stasiun bumi yang tersedia untuk umum yang tercantum di atas,AWS Ground Station menawarkan Antena Khusus. Antena Khusus adalah sistem antena yang dibuat khusus yang AWS mengelola atas nama Anda. Antena Khusus tidak terbatas pada lokasi stasiun AWS Ground Station bumi yang ada dan dapat dibangun dengan kemampuan di luar stasiun bumi umum seperti yang dijelaskan dalamAWS Ground Station Kemampuan Situs. Lokasi dan kemampuan Antena Khusus tidak diungkapkan kepada publik.

Untuk informasi lebih lanjut tentang Antena Khusus, lihatAWS Ground Station Antena Khusus. Untuk mempelajari lebih lanjut atau untuk memulai dengan Antena Khusus, hubungi AWS Dukungan melalui. AWS Support Center Console

Melihat antena di stasiun bumi

Setiap lokasi stasiun bumi memiliki satu atau lebih antena. Anda dapat melihat antena di stasiun bumi dengan menggunakan ListAntennasAPI. API ini mengembalikan antena di stasiun bumi tertentu, termasuk nama masing-masing antena.

Informasi antena berguna bila dikombinasikan dengan ListGroundStationReservationsAPI untuk memahami kapasitas dan ketersediaan di stasiun bumi. Untuk informasi selengkapnya tentang melihat reservasi, lihatLihat reservasi stasiun bumi.

Untuk meneleponListAntennas, Anda harus memiliki satelit yang terpasang ke stasiun bumi atau memiliki izin ephemeris elevasi azimuth untuk stasiun bumi. Untuk informasi selengkapnya, lihat Berikan data ephemeris elevasi azimuth.

Contoh: Daftar antena di stasiun bumi

Contoh berikut mencantumkan semua antena di stasiun bumi menggunakan AWS SDK for Python (Boto3).

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The ground station ID to list antennas for.
# Use the ListGroundStations API to find available ground station IDs.
ground_station_id = "Ohio 1"

# List all antennas at a ground station.
# This is useful for understanding the capacity of a ground station
# and for planning multi-antenna operations.
print(f"Listing antennas for ground station '{ground_station_id}'...")

paginator = ground_station_client.get_paginator("list_antennas")
page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)
```

```
for page in page_iterator:
    for antenna in page["antennaList"]:
        print(f"  Antenna: {antenna['antennaName']}")
        print(f"    Ground Station: {antenna['groundStationName']}")
        print(f"    Region: {antenna['region']}")
        print()
```

Lihat reservasi stasiun bumi

Anda dapat melihat reservasi di seluruh antena di stasiun bumi dengan menggunakan API. ListGroundStationReservations Reservasi mewakili blok waktu pada antena, termasuk kontak terjadwal Anda. AWS Ground Station Antena Khususpelanggan juga melihat jendela pemeliharaan.

Informasi ini membantu Anda memahami ketersediaan antena saat merencanakan jadwal kontak dan memberikan visibilitas ke apa yang terjadi pada antena di stasiun bumi.

Reservasi daftar

Untuk membuat daftar reservasi, hubungi ListGroundStationReservationsdengan pengenal stasiun bumi dan rentang waktu. API mengembalikan reservasi di semua antena di stasiun bumi dalam jangka waktu yang ditentukan.

Reservasi yang Anda lihat tergantung pada tingkat akses Anda:

- AWS Ground Station Pelanggan publik — Anda hanya dapat melihat reservasi kontak Anda sendiri. Jendela pemeliharaan dan kontak yang dimiliki oleh akun lain tidak termasuk.
- AWS Ground Station Pelanggan Antena Khusus — Anda dapat melihat semua pemesanan di Antena Khusus Anda, termasuk jendela pemeliharaan dan kontak yang dijadwalkan oleh akun lain. Pengidentifikasi kontak hanya disertakan untuk kontak yang Anda miliki. Untuk informasi selengkapnya, lihat AWS Ground Station Antena Khusus.

Jenis reservasi

Setiap reservasi memiliki jenis yang menunjukkan untuk apa waktu antena digunakan:

- Kontak - Reservasi kontak mewakili waktu antena yang disediakan untuk komunikasi satelit. Waktu mulai dan akhir reservasi mencerminkan reservasi antena penuh, termasuk waktu pra-pass dan pasca-lulus, bukan hanya jendela pass satelit.

- Pemeliharaan - Reservasi pemeliharaan mewakili periode waktu ketika antena tidak tersedia karena pemeliharaan. Reservasi pemeliharaan termasuk a `maintenanceType` yang menunjukkan apakah pemeliharaan direncanakan atau tidak direncanakan.

Contoh kode

Contoh berikut mencantumkan reservasi di stasiun bumi selama 7 hari ke depan menggunakan AWS SDK for Python (Boto3), termasuk pemfilteran berdasarkan jenis reservasi.

```
import boto3
from datetime import datetime, timezone, timedelta

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The ground station ID to list reservations for
ground_station_id = "Ohio 1"

# Define the time range to query. Reservations include both your
# scheduled contacts and maintenance windows at the ground station.
start_time = datetime.now(timezone.utc)
end_time = start_time + timedelta(days=7)

# List all reservations at a ground station for the next 7 days.
# You can filter by reservation type to see only contacts or
# only maintenance windows.
print(f"Listing reservations for ground station '{ground_station_id}'...")
print(f"Time range: {start_time} to {end_time}")

paginator = ground_station_client.get_paginator("list_ground_station_reservations")
page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    startTime=start_time,
    endTime=end_time,
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for reservation in page["reservationList"]:
```

```
        reservation_type = reservation["reservationType"]
        antenna_name = reservation["antennaName"]
        res_start = reservation["startTime"]
        res_end = reservation["endTime"]

        print(f"  Type: {reservation_type}")
        print(f"    Antenna: {antenna_name}")
        print(f"    Start: {res_start}")
        print(f"    End: {res_end}")

        details = reservation["reservationDetails"]
        if "contact" in details:
            contact_id = details["contact"].get("contactId", "N/A")
            print(f"    Contact ID: {contact_id}")
        elif "maintenance" in details:
            maintenance_type = details["maintenance"]["maintenanceType"]
            print(f"    Maintenance Type: {maintenance_type}")

        print()

# For Dedicated Antenna customers, you can also filter to show only maintenance windows
print("Listing only maintenance reservations...")

page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    startTime=start_time,
    endTime=end_time,
    reservationTypes=["MAINTENANCE"],
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for reservation in page["reservationList"]:
        maintenance_type = reservation["reservationDetails"]["maintenance"][
            "maintenanceType"
        ]
        print(
            f"  {maintenance_type} maintenance on {reservation['antennaName']}: "
            f"{reservation['startTime']} to {reservation['endTime']}"
        )
```

AWS Ground Station topeng situs

Setiap lokasi AWS Ground Station antena memiliki topeng situs terkait. Masker ini memblokir antena di lokasi itu agar tidak mentransmisikan atau menerima saat menunjuk ke beberapa arah, biasanya dekat dengan cakrawala. Topeng dapat memperhitungkan:

- Fitur medan geografis yang mengelilingi antena — Misalnya, ini termasuk hal-hal seperti gunung atau bangunan, yang akan memblokir sinyal frekuensi radio (RF) atau mencegah transmisi.
- Interferensi Frekuensi Radio (RFI) — Ini memengaruhi kemampuan untuk menerima (sumber RFI eksternal yang memengaruhi sinyal downlink ke antena AWS Ground Station) dan transmisi (sinyal RF yang ditransmisikan oleh antena AWS Ground Station berdampak buruk pada penerima eksternal).
- Otorisasi hukum — Otorisasi situs lokal untuk mengoperasikan AWS Ground Station di setiap wilayah dapat mencakup pembatasan tertentu, seperti sudut ketinggian minimum untuk transmisi.

Masker situs ini dapat berubah seiring waktu. Misalnya, bangunan baru dapat dibangun di dekat lokasi antena, sumber RFI dapat berubah, atau otorisasi hukum dapat diperbarui dengan pembatasan yang berbeda. Masker situs AWS Ground Station tersedia untuk Anda berdasarkan perjanjian non-disclosure (NDA).

Masker khusus pelanggan

Selain masker situs AWS Ground Station di setiap situs, Anda mungkin memiliki masker tambahan karena pembatasan otorisasi hukum Anda sendiri untuk berkomunikasi dengan satelit Anda di wilayah tertentu. Masker semacam itu dapat dikonfigurasi di AWS Ground case-by-case Station untuk memastikan kepatuhan saat menggunakan AWS Ground Station untuk berkomunikasi dengan satelit ini. Hubungi tim AWS Ground Station untuk detailnya.

Dampak masker situs pada waktu kontak yang tersedia

Ada dua jenis masker situs: topeng situs uplink (kirim), dan topeng situs downlink (terima).

Saat mencantumkan waktu kontak yang tersedia menggunakan ListContacts operasi, AWS Ground Station akan mengembalikan waktu visibilitas berdasarkan kapan satelit Anda akan naik di atas dan disetel di bawah downlink mask. Waktu kontak yang tersedia didasarkan pada jendela visibilitas downlink mask ini. Ini memastikan bahwa Anda tidak memesan waktu ketika satelit Anda berada di bawah downlink mask.

Masker situs Uplink tidak diterapkan pada waktu kontak yang tersedia, bahkan jika Profil Misi menyertakan Konfigurasi Uplink Antena di tepi aliran data. Ini memungkinkan Anda untuk menggunakan semua waktu kontak yang tersedia untuk downlink, bahkan jika uplink mungkin tidak tersedia untuk sebagian waktu itu karena topeng situs uplink. Namun, sinyal uplink mungkin tidak ditransmisikan untuk beberapa atau sepanjang waktu yang disediakan untuk kontak satelit. Anda bertanggung jawab untuk menghitung masker uplink yang disediakan saat menjadwalkan transmisi uplink.

Bagian kontak yang tidak tersedia untuk uplink bervariasi tergantung pada lintasan satelit selama kontak, relatif terhadap topeng situs uplink di lokasi antena. Di daerah di mana topeng situs uplink dan downlink serupa, durasi ini biasanya akan pendek. Di wilayah lain, di mana topeng uplink mungkin jauh lebih tinggi daripada topeng situs downlink, ini dapat mengakibatkan sebagian besar, atau bahkan semua, durasi kontak tidak tersedia untuk uplink. Waktu kontak penuh ditagih kepada Anda, bahkan jika sebagian dari waktu yang dipesan tidak tersedia untuk uplink.

AWS Ground Station Kemampuan Situs

Untuk menyederhanakan pengalaman Anda, AWS Ground Station tentukan seperangkat kemampuan umum untuk jenis antena dan kemudian menyebarkan beberapa antena ke lokasi stasiun bumi. Bagian dari langkah orientasi memastikan satelit Anda kompatibel dengan jenis antena di lokasi tertentu. Ketika Anda memesan kontak, Anda secara tidak langsung menentukan jenis antena yang digunakan. Ini memastikan pengalaman Anda di lokasi stasiun bumi tertentu tetap sama dari waktu ke waktu terlepas dari antena mana yang digunakan. Kinerja spesifik kontak Anda akan bervariasi karena berbagai masalah lingkungan seperti cuaca di lokasi.

Saat ini, semua situs mendukung kemampuan berikut:

> **Note**
>
> Setiap baris dalam tabel berikut menunjukkan jalur komunikasi independen, kecuali dinyatakan lain. Baris duplikat ada untuk mencerminkan kemampuan multi-saluran kami yang memungkinkan beberapa jalur komunikasi digunakan secara bersamaan.

Jenis Kemampuan	Rentang Frekuensi	Rentang Bandwidth	Polarisasi	Nama Umum	Catatan
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP	Downlink pita lebar X-band	Kemampuan ini membutuhkan penggunaan AWS Ground Station Agen. Kemampuan ini tidak didukung di Alaska 1 atau Punta Arenas 1. Bandwidth agregat tidak boleh melebihi 400 MHz per polarisasi di setiap lokasi. Semua rentang frekuensi yang digunakan harus tidak tumpang tindih.
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antena-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antena-downlink	2200 - 2290 MHz	Hingga 40 MHz	RHCP	Downlink S-band	Hanya satu polarisasi yang dapat

Jenis Kemampuan	Rentang Frekuensi	Rentang Bandwidth	Polarisasi	Nama Umum	Catatan
antena-downlink	2200 - 2290 MHz	Hingga 40 MHz	LHCP		digunakan pada satu waktu
antena-downlink	7750 - 8500 MHz	Hingga 40 MHz	RHCP	Downlink narrowband X-band	Hanya satu polarisasi yang dapat digunakan pada satu waktu
antena-downlink	7750 - 8500 MHz	Hingga 40 MHz	LHCP		
antena-uplink	2025 - 2110 MHz	Hingga 40 MHz	RHCP	Tautan atas S-band	Hanya satu polarisasi yang dapat digunakan pada satu waktu EIRP 20-50 dBW
antena-uplink	2025 - 2110 MHz	Hingga 40 MHz	LHCP		
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	RHCP	Gema uplink	Cocokkan pembatasan antena-uplink
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	LHCP		
antenna-downlink-demod-decode	7750 - 8500 MHz	Hingga 500 MHz	RHCP	Downlink X-band yang didemodulasi dan diterjemahkan	
antenna-downlink-demod-decode	7750 - 8500 MHz	Hingga 500 MHz	LHCP		

Jenis Kemampuan	Rentang Frekuensi	Rentang Bandwidth	Polarisasi	Nama Umum	Catatan
pelacakan	N/A	N/A	N/A	N/A	Support untuk pelacakan otomatis dan pelacakan program

* RHCP = polarisasi melingkar tangan kanan, dan LHCP = polarisasi melingkar tangan kiri. Untuk informasi lebih lanjut tentang polarisasi, lihat Polarisasi melingkar.

Memahami caranya AWS Ground Station menggunakan ephemerides

Ephemeris, ephemerides jamak, adalah file atau struktur data yang menyediakan lintasan objek astronomi. Secara historis, file ini hanya mengacu pada data tabular tetapi, secara bertahap, telah mengarahkan ke berbagai file data yang menunjukkan lintasan pesawat ruang angkasa.

API Ephemeris memungkinkan ephemerides khusus untuk diunggah untuk digunakan dengan satelit.AWS Ground Station Ephemerides ini mengesampingkan ephemerides default dari (lihat:). Space-TrackData ephemeris standarAWS Ground Station mendukung data ephemeris dalam Orbit Ephemeris Message (OEM), elemen dua baris (TLE), dan format elevasi azimuth. Untuk akurasi tertinggi dan cakupan rentang waktu paling fleksibel, berikan data ephemeris dalam format OEM (lihat). Menyediakan data ephemeris OEM

AWS Ground Station menggunakan data ephemeris untuk menentukan kapan kontak tersedia berdasarkan ephemeris yang disediakan dan antena perintah yang benar dalam jaringan.AWS Ground StationSecara default, tidak ada tindakan yang diperlukan untuk menyediakan AWS Ground Station ephemerides jika satelit Anda memiliki ID NORAD yang ditetapkan.

Mengunggah ephemerides khusus dapat meningkatkan kualitas pelacakan, menangani operasi awal di mana tidak ada Space-Trackephemerides yang tersedia, dan memperhitungkan manuver.AWS Ground Station

Atau,AWS Ground Station mendukung format elevasi azimuth, yang memungkinkan Anda untuk secara langsung menentukan arah penunjuk antena tanpa memberikan informasi orbital satelit. Ini berguna untuk skenario di mana penunjuk antena yang tepat diperlukan karena informasi lintasan satelit tidak tepat atau tidak diketahui.

Topik

- Data ephemeris standar
- Berikan data ephemeris khusus
- Cadangan kontak dengan ephemeris khusus
- Memahami ephemeris mana yang digunakan
- Dapatkan ephemeris saat ini untuk satelit
- Kembalikan ke data ephemeris default

Data ephemeris standar

Secara default,AWS Ground Station menggunakan data yang tersedia untuk umum dari Space-Track, dan tidak ada tindakan yang diperlukan untuk memasok AWS Ground Station ephemerides default ini. Ephemerides ini adalah set elemen dua baris (TLE) yang terkait dengan ID NORAD satelit Anda. Semua ephemerides default memiliki prioritas. 0 Akibatnya, mereka akan diganti, selalu, oleh ephemerides khusus yang tidak kedaluwarsa yang diunggah melalui API ephemeris, yang harus selalu memiliki prioritas, atau lebih besar. 1

Satelit tanpa ID NORAD harus mengunggah data ephemeris khusus ke.AWS Ground Station Misalnya, satelit yang baru saja diluncurkan atau yang sengaja dihilangkan dari Space-Trackkatalog tidak akan memiliki ID NORAD dan perlu mengunggah ephemerides khusus. Untuk informasi selengkapnya tentang penyediaan data ephemeris khusus, lihat: Menyediakan Data Ephemeris Kustom.

Berikan data ephemeris khusus

> **Important**
>
> API ephemeris saat ini dalam status Pratinjau

Akses ke API Ephemeris disediakan hanya sesuai kebutuhan. Jika Anda memerlukan kemampuan untuk mengunggah data ephemeris khusus, silakan buka AWS Dukungan tiket melalui. AWS Support Center Console Tim kami akan bekerja dengan Anda untuk mengaktifkan kemampuan ini untuk kebutuhan spesifik Anda.

Gambaran umum

API Ephemeris memungkinkan ephemerides khusus untuk diunggah untuk digunakan dengan satelit.AWS Ground Station Ephemerides ini mengesampingkan ephemerides default dari (lihat:). Space-TrackData ephemeris standar Kami mendukung penerimaan data ephemeris dalam Orbit Ephemeris Message (OEM), elemen dua baris (TLE), dan format elevasi azimuth.

AWS Ground Station memperlakukan ephemerides sebagai Data Penggunaan Individual. Jika Anda menggunakan fitur opsional ini, AWS akan menggunakan data ephemeris Anda untuk memberikan dukungan pemecahan masalah.

Mengunggah ephemerides khusus dapat meningkatkan kualitas pelacakan, menangani operasi di mana tidak ada Space-Trackephemerides yang tersedia, dan memperhitungkan manuver.AWS Ground Station

Untuk memecahkan masalah ephemeris yang tidak valid, lihat: Memecahkan masalah ephemerides yang tidak valid

Contoh: Menggunakan ephemerides yang disediakan pelanggan dengan AWS Ground Station

Untuk petunjuk lebih rinci tentang penggunaan ephemerides yang disediakan pelanggan AWS Ground Station, lihat Menggunakan ephemerides yang disediakan pelanggan dengan dan repositori terkait aws- -groundstation-cpe.AWS Ground Station GitHub samples/aws

Berikan data ephemeris TLE

> **Important**
>
> API ephemeris saat ini dalam status Pratinjau

Akses ke API Ephemeris disediakan hanya sesuai kebutuhan. Jika Anda memerlukan kemampuan untuk mengunggah data ephemeris khusus, silakan buka AWS Dukungan tiket melalui. AWS Support Center Console Tim kami akan bekerja dengan Anda untuk mengaktifkan kemampuan ini untuk kebutuhan spesifik Anda.

Gambaran umum

Two-line set elemen (TLE) adalah format standar untuk menggambarkan orbit satelit. API Ephemeris memungkinkan ephemerides TLE diunggah untuk digunakan dengan satelit.AWS Ground Station Ephemerides ini mengesampingkan ephemerides default dari (lihat:). Space-TrackData ephemeris standar

AWS Ground Station memperlakukan ephemerides sebagai Data Penggunaan Individual. Jika Anda menggunakan fitur opsional ini, AWS akan menggunakan data ephemeris Anda untuk memberikan dukungan pemecahan masalah.

Mengunggah ephemerides TLE khusus dapat meningkatkan kualitas pelacakan, menangani operasi awal di mana tidak ada Space-Trackephemerides yang tersedia, dan memperhitungkan manuver.AWS Ground Station Namun, akurasi TLE dibatasi oleh model propagasi dan menurun

seiring waktu jauh dari zaman. Untuk data ephemeris dengan kesetiaan yang lebih tinggi dalam rentang waktu yang lebih lama, pertimbangkan untuk menggunakan format OEM (lihat). Menyediakan data ephemeris OEM

> **Note**
>
> Saat memberikan ephemeris khusus sebelum nomor katalog satelit ditetapkan untuk satelit Anda, Anda dapat menggunakan `00000` untuk bidang nomor katalog satelit TLE, dan `000` untuk bagian nomor peluncuran bidang penunjuk internasional TLE (misalnya `24000A` untuk kendaraan yang diluncurkan pada tahun 2024).
> Untuk informasi selengkapnya tentang format TTLE, lihat kumpulan Two-line elemen.

Alpha-5 nomor katalog satelit di TLE

Two-Line Set elemen (TLE) menggunakan format lebar tetap di mana bidang nomor katalog satelit (kolom 3-7 baris 1 dan 2) dibatasi hingga 5 digit, mendukung nilai maksimum 99.999. Untuk mewakili nomor katalog satelit di luar batas ini, Angkatan Luar Angkasa Amerika Serikat mengembangkan skema Alpha-5pengkodean. Alpha-5 menggantikan digit pertama bidang nomor katalog satelit dengan karakter alfabet (A—Z, tidak termasuk I dan O untuk menghindari kebingungan dengan angka 1 dan 0). Ini memperluas jangkauan nomor katalog satelit yang dapat direpresentasikan dari 100.000 hingga 339.999 dalam format TLE lebar tetap yang ada.

Alpha-5 adalah murni pengkodean tampilan untuk format teks TLE. Nomor katalog satelit numerik yang mendasarinya tidak berubah. `noradSatelliteID`Bidang di AWS Ground Station API tetap merupakan integer — respons API seperti `ListSatellites` dan `GetSatellite` terus mengembalikan nomor katalog satelit numerik, bukan formulir yang Alpha-5 dikodekan. Alpha-5 pengkodean hanya relevan dalam baris teks TLE itu sendiri.

AWS Ground Station mendukung ephemerides TLE yang mengandung nomor satelit yang Alpha-5 dikodekan. Saat Anda mengunggah TLE melalui CreateEphemeristindakan, mem-parsing nomor satelit yang Alpha-5 dikodekan AWS Ground Station dengan benar dan menyelesaikannya ke nomor katalog satelit integer yang sesuai secara internal. Satelit dengan nomor katalog satelit dalam Alpha-5 kisaran (100.000—339.999) didukung penuh.

Berikut ini adalah contoh TLE yang ditetapkan untuk satelit dengan katalog satelit nomor 100.000. Alpha-5 Pengkodean mewakili ini seperti `A0000` pada kolom 3-7 dari setiap baris TLE:

```
[
```

```
    {
        "tleLine1": "1 A0000U 26001A   26120.50000000  .00000072  00000-0  24500-4 0
 9991",
        "tleLine2": "2 A0000  97.4500  85.2300 0012345  45.6789 314.5678 15.20000000
 10013",
        "validTimeRange": {
            "startTime": 1777680000,
            "endTime": 1778284800
        }
    }
]
```

Untuk informasi lebih lanjut tentang skema Alpha-5 pengkodean, lihat Space-Track Alpha-5dokumentasi.

Membuat ephemeris TLE

Ephemeris TLE dapat dibuat menggunakan CreateEphemeristindakan di API.AWS Ground Station Tindakan ini akan mengunggah ephemeris menggunakan data baik di badan permintaan atau dari bucket S3 yang ditentukan.

Penting untuk dicatat bahwa mengunggah ephemeris menyetel ephemeris `VALIDATING` dan memulai alur kerja asinkron yang akan memvalidasi dan menghasilkan kontak potensial dari ephemeris Anda. Hanya setelah ephemeris melewati alur kerja ini dan menjadi `ENABLED` akan digunakan untuk kontak. Anda harus melakukan polling DescribeEphemerisuntuk status ephemeris atau menggunakan CloudWatch peristiwa untuk melacak perubahan status ephemeris.

Untuk memecahkan masalah ephemeris yang tidak valid, lihat: Memecahkan masalah ephemerides yang tidak valid

Contoh: Buat elemen dua baris (TLE) set ephemeris melalui API

AWS SDK, dan CLI dapat digunakan untuk mengunggah elemen dua baris (TLE) yang disetel AWS Ground Station ephemeris melalui panggilan. CreateEphemeris Ephemeris ini akan digunakan sebagai pengganti data ephemeris default untuk satelit (lihat). Data ephemeris standar Contoh ini menunjukkan bagaimana melakukan ini menggunakan AWS SDK for Python (Boto3).

Set TLE adalah objek berformat JSON yang merangkai satu atau lebih TLE bersama-sama untuk membangun lintasan kontinu. TLE dalam set TLE harus membentuk himpunan kontinu yang dapat kita gunakan untuk membangun lintasan (yaitu tidak ada celah waktu antara TLE dalam set TLE). Contoh set TLE ditunjukkan di bawah ini:

```
[
    {
        "tleLine1": "1 25994U 99068A   20318.54719794  .00000075  00000-0  26688-4 0
 9997",
        "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
        "validTimeRange": {
            "startTime": 12345,
            "endTime": 12346
        }
    },
    {
        "tleLine1": "1 25994U 99068A   20318.54719794  .00000075  00000-0  26688-4 0
 9997",
        "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
        "validTimeRange": {
            "startTime": 12346,
            "endTime": 12347
        }
    }
]
```

> **Note**
>
> Rentang waktu TLE dalam set TLE harus sama persis untuk menjadi lintasan berkelanjutan yang valid.

Satu set TLE dapat diunggah melalui klien AWS Ground Station boto3 sebagai berikut:

```
import boto3
from datetime import datetime, timedelta, timezone

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create TLE ephemeris
tle_ephemeris = ground_station_client.create_ephemeris(
    name="Example Ephemeris",
    satelliteId="2e925701-9485-4644-b031-EXAMPLE01",
    enabled=True,
```

```
    expirationTime=datetime.now(timezone.utc) + timedelta(days=3),
    priority=2,
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   20318.54719794  .00000075  00000-0
 26688-4 0  9997",
                    "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
            ]
        }
    },
)

print(f"Created TLE ephemeris with ID: {tle_ephemeris['ephemerisId']}")
```

Panggilan ini akan mengembalikan ephemerisid yang dapat digunakan untuk mereferensikan ephemeris di masa depan. Misalnya, kita dapat menggunakan ephemerisid yang disediakan dari panggilan di atas untuk melakukan polling untuk status ephemeris:

```
import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# First, create a TLE ephemeris
print("Creating TLE ephemeris...")

tle_ephemeris = ground_station_client.create_ephemeris(
    name="Example TLE Ephemeris for Description",
    satelliteId="2e925701-9485-4644-b031-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=3),
    priority=2,
    ephemeris={
```

```
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   20318.54719794  .00000075  00000-0
 26688-4 0  9997",
                    "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
            ]
        }
    },
)

ephemeris_id = tle_ephemeris["ephemerisId"]
print(f"Created TLE ephemeris with ID: {ephemeris_id}")

# Describe the ephemeris immediately to check initial status
print("Describing ephemeris...")

response = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)

print(f"Ephemeris ID: {response['ephemerisId']}")
print(f"Name: {response['name']}")
print(f"Status: {response['status']}")
```

Contoh respons dari DescribeEphemeristindakan disediakan di bawah ini

```
{
  "creationTime": 1620254718.765,
  "enabled": true,
  "name": "Example Ephemeris",
  "ephemerisId": "fde41049-14f7-413e-bd7b-EXAMPLE01",
  "priority": 2,
  "status": "VALIDATING",
  "suppliedData": {
    "tle": {
      "ephemerisData": "[{\"tleLine1\": \"1 25994U 99068A   20318.54719794  .00000075
 00000-0  26688-4 0  9997\",\"tleLine2\": \"2 25994  98.2007  30.6589 0001234  89.2782
```

```
  18.9934 14.57114995111906\",\"validTimeRange\": {\"startTime\": 1620254712000,
\"endTime\": 1620859512000}}]"
    }
  }
}
```

Disarankan untuk melakukan polling DescribeEphemerisrute atau menggunakan CloudWatch peristiwa untuk melacak status ephemeris yang diunggah karena harus melalui alur kerja validasi asinkron sebelum disetel ke dan menjadi dapat digunakan untuk ENABLED menjadwalkan dan mengeksekusi kontak.

Perhatikan bahwa ID NORAD di semua TLE dalam set TLE, 25994 dalam contoh di atas, harus cocok dengan ID NORAD yang telah ditetapkan satelit Anda dalam database. Space-Track

Contoh: Mengunggah data ephemeris TLE dari bucket S3

Dimungkinkan juga untuk mengunggah file ephemeris TLE langsung dari bucket S3 dengan menunjuk ke bucket dan kunci objek.AWS Ground Station akan mengambil objek atas nama Anda. Informasi tentang enkripsi data saat istirahat AWS Ground Station dirinci dalam:Enkripsi data saat istirahat untuk AWS Ground Station.

Di bawah ini adalah contoh mengunggah file ephemeris TLE dari bucket S3

```
import boto3
from datetime import datetime, timedelta, timezone
import json

# Create AWS clients
s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

# Define S3 bucket and key
bucket_name = "ephemeris-bucket"
object_key = "test_data.tle"

# Create sample TLE set data
# Note: For actual satellites, use real TLE data from sources like Space-Track
tle_set_data = [
    {
        "tleLine1": "1 25994U 99068A   20318.54719794  .00000075  00000-0  26688-4 0
 9997",
```

```
        "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc),
            "endTime": datetime.now(timezone.utc) + timedelta(days=3),
        },
    },
    {
        "tleLine1": "1 25994U 99068A   20321.54719794  .00000075  00000-0  26688-4 0
 9998",
        "tleLine2": "2 25994  98.2007  33.6589 0001234  89.2782  18.9934
 14.57114995112342",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc) + timedelta(days=3),
            "endTime": datetime.now(timezone.utc) + timedelta(days=7),
        },
    },
]

# Convert to JSON string for upload
tle_json = json.dumps(tle_set_data, indent=2)

# Upload sample TLE data to S3
print(f"Uploading TLE set data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name, Key=object_key, Body=tle_json, ContentType="application/json"
)
print("TLE set data uploaded successfully to S3")
print(f"Uploaded {len(tle_set_data)} TLE entries covering 7 days")

# Create TLE ephemeris from S3
print("Creating TLE ephemeris from S3...")

s3_tle_ephemeris = ground_station_client.create_ephemeris(
    name="2022-11-05 S3 TLE Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,
    ephemeris={"tle": {"s3Object": {"bucket": bucket_name, "key": object_key}}},
)
```

```
print(f"Created TLE ephemeris with ID: {s3_tle_ephemeris['ephemerisId']}")
```

Menyediakan data ephemeris OEM

> **Important**
>
> API ephemeris saat ini dalam status Pratinjau

Akses ke API Ephemeris disediakan hanya sesuai kebutuhan. Jika Anda memerlukan kemampuan untuk mengunggah data ephemeris khusus, silakan buka AWS Dukungan tiket melalui. AWS Support Center Console Tim kami akan bekerja dengan Anda untuk mengaktifkan kemampuan ini untuk kebutuhan spesifik Anda.

Gambaran umum

Orbit Ephemeris Message (OEM) adalah format standar untuk mewakili data lintasan pesawat ruang angkasa. API Ephemeris memungkinkan ephemerides OEM diunggah untuk digunakan dengan satelit.AWS Ground Station Ephemerides ini mengesampingkan ephemerides default dari (lihat:). Space-TrackData ephemeris standar

AWS Ground Station memperlakukan ephemerides sebagai Data Penggunaan Individual. Jika Anda menggunakan fitur opsional ini,AWS akan menggunakan data ephemeris Anda untuk memberikan dukungan pemecahan masalah.

Mengunggah ephemerides OEM khusus dapat meningkatkan kualitas pelacakan, menangani operasi awal di mana tidak ada Space-Trackephemerides yang tersedia, dan memperhitungkan manuver.AWS Ground Station

> **Note**
>
> Saat memberikan ephemeris khusus sebelum nomor katalog satelit ditetapkan untuk satelit Anda, Anda dapat menggunakan `satelliteId` untuk `OBJECT_ID` bagian OEM.
> Untuk informasi lebih lanjut tentang format OEM, lihatFormat ephemer OEM.

Format ephemer OEM

AWS Ground Station memproses Ephemerides yang Disediakan Pelanggan OEM sesuai dengan standar CCSDS dengan beberapa batasan tambahan. File OEM harus dalam format KVN. Tabel

berikut menguraikan bidang yang berbeda dalam OEM dan bagaimana AWS Ground Station perbedaannya dari standar CCSDS.

Bagian	Bidang	CCSDS diperlukan	AWS Ground Station diperlukan	Catatan
Header	CCSDS_OEM_VERS	Ya	Ya	Nilai yang dibutuhkan: 2.0
	MENGOMENTARI	Tidak	Tidak	
	KLASIFIKASI	Tidak	Tidak	
	CREATION_DATE	Ya	Ya	
	PENCETUS	Ya	Ya	
	MESSAGE_ID	Tidak	Tidak	
Metadata	META_START	Ya	Ya	
	MENGOMENTARI	Tidak	Tidak	
	OBJECT_NAME	Ya	Ya	
	OBJECT_ID	Ya	Ya	
	CENTER_NAME	Ya	Ya	Nilai yang dibutuhkan: Bumi
	REF_FRAME	Ya	Ya	Nilai yang diterima: EME2000, ITRF2000

Bagian	Bidang	CCSDS diperlukan	AWS Ground Station diperlukan	Catatan
	REF_FRAME_EPOCH	Tidak	Tidak didukung*	Tidak diperlukan karena Ref_frames yang diterima memiliki epoch implisit
	TIME_SISTEM	Ya	Ya	Nilai yang dibutuhkan: UTC
	START_TIME	Ya	Ya	
	DAPAT DIGUNAKAN_START_TIME	Tidak	Tidak	
	DAPAT DIGUNAKAN_STOP_TIME	Tidak	Tidak	
	BERHENTI_WAKTU	Ya	Ya	
	INTERPOLASI	Tidak	Ya	Diperlukan sehingga AWS Ground Station dapat menghasilkan sudut penunjuk yang akurat untuk kontak.

Bagian	Bidang	CCSDS diperlukan	AWS Ground Station diperlukan	Catatan
	INTERPOLASI_DERAJAT	Tidak	Ya	Diperlukan sehingga AWS Ground Station dapat menghasilkan sudut penunjuk yang akurat untuk kontak. Gelar yang ditentukan akan digunakan jika memungkinkan, tetapi tingkat yang lebih rendah akan digunakan jika tidak ada cukup data di segmen tersebut.
	META_STOP	Ya	Ya	
Data	X	Ya	Ya	Diwakili dalam `km`
	Y	Ya	Ya	Diwakili dalam `km`
	Z	Ya	Ya	Diwakili dalam `km`
	X_DOT	Ya	Ya	Diwakili dalam `km/s`

Bagian	Bidang	CCSDS diperlukan	AWS Ground Station diperlukan	Catatan
	Y_DOT	Ya	Ya	Diwakili dalam `km/s`
	Z_DOT	Ya	Ya	Diwakili dalam `km/s`
	X_DDOT	Tidak	Tidak	Diwakili dalam `km/s^2`
	Y_DDOT	Tidak	Tidak	Diwakili dalam `km/s^2`
	Z_DDOT	Tidak	Tidak	Diwakili dalam `km/s^2`
Matriks kovarians	COVARIANCE_START	Tidak	Tidak	
	EPOCH	Tidak	Tidak	
	COV_REF_FRAME	Tidak	Tidak	
	KOVARIANCE_STOP	Tidak	Tidak	

* Jika ada baris yang tidak didukung oleh AWS Ground Station termasuk dalam OEM yang disediakan, OEM akan gagal validasi.

Penyimpangan penting dari standar CCSDS adalah:AWS Ground Station

- `CCSDS_OEM_VERS`diperlukan untuk menjadi`2.0`.
- `REF_FRAME`diperlukan untuk menjadi salah satu `EME2000` atau `ITRF2000`.
- `REF_FRAME_EPOCH`tidak didukung oleh AWS Ground Station.

- `CENTER_NAME`diperlukan untuk menjadi`Earth`.
- `TIME_SYSTEM`diperlukan untuk menjadi`UTC`.
- `INTERPOLATION`dan `INTERPOLATION_DEGREE` keduanya diperlukan untuk ephemeris yang disediakan AWS Ground Station pelanggan.
- AWS Ground Station menyimpang dari CCSDS 5.2.4.7 dengan mengizinkan blok data OEM yang tidak mengandung catatan data ephemeris yang cukup untuk melakukan interpolasi pada yang ditentukan. `INTERPOLATION_DEGREE` Dalam hal ini,AWS Ground Station akan menggunakan tingkat interpolasi tertinggi yang mungkin kurang dari atau sama dengan yang ditentukan. `INTERPOLATION_DEGREE`

Contoh OEM ephemeris dalam format KVN

Berikut ini adalah contoh terpotong dari ephemeris OEM dalam format KVN untuk satelit penyiar publik. JPSS-1

```
CCSDS_OEM_VERS = 2.0

COMMENT Orbit data are consistent with planetary ephemeris DE-430

CREATION_DATE  = 2024-07-22T05:20:59
ORIGINATOR     = Raytheon-JPSS/CGS

META_START
OBJECT_NAME          = J1
OBJECT_ID            = 2017-073A
CENTER_NAME          = Earth
REF_FRAME            = EME2000
TIME_SYSTEM          = UTC
START_TIME           = 2024-07-22T00:00:00.000000
STOP_TIME            = 2024-07-22T00:06:00.000000
INTERPOLATION        = Lagrange
INTERPOLATION_DEGREE = 5
META_STOP

2024-07-22T00:00:00.000000   5.905147360000000e+02  -1.860082793999999e+03
  -6.944807075000000e+03  -5.784245796000000e+00   4.347501391999999e+00
 -1.657256863000000e+00
2024-07-22T00:01:00.000000   2.425572045154201e+02  -1.595860765983339e+03
  -7.030938457373539e+03  -5.810660250794190e+00   4.457103652219009e+00
 -1.212889340333023e+00
```

```
2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
  -7.090262617183503e+03  -5.814973972202444e+00   4.549739160042560e+00
 -7.639633689161465e-01
2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
  -7.122556683227951e+03  -5.797176562437553e+00   4.625064829516728e+00
 -3.121687831090774e-01
2024-07-22T00:04:00.000000  -8.015427368657785e+02  -7.709137891269565e+02
  -7.127699477194810e+03  -5.757338007808417e+00   4.682800822515077e+00
 1.407953645161997e-01
2024-07-22T00:05:00.000000  -1.145240083085062e+03  -4.886583601179489e+02
  -7.105671911254255e+03  -5.695608435738609e+00   4.722731329786999e+00
 5.932259682105052e-01
2024-07-22T00:06:00.000000  -1.484582479061495e+03  -2.045451985605701e+02
  -7.056557069672793e+03  -5.612218005854990e+00   4.744705579872771e+00
 1.043421397392599e+00
```

Membuat ephemeris OEM

Ephemeris OEM dapat dibuat menggunakan CreateEphemeristindakan di API.AWS Ground Station Tindakan ini akan mengunggah ephemeris menggunakan data baik di badan permintaan atau dari bucket S3 yang ditentukan.

Penting untuk dicatat bahwa mengunggah ephemeris menyetel ephemeris `VALIDATING` dan memulai alur kerja asinkron yang akan memvalidasi dan menghasilkan kontak potensial dari ephemeris Anda. Hanya setelah ephemeris melewati alur kerja ini dan menjadi `ENABLED` akan digunakan untuk kontak. Anda harus melakukan polling DescribeEphemerisuntuk status ephemeris atau menggunakan CloudWatch peristiwa untuk melacak perubahan status ephemeris.

Untuk memecahkan masalah ephemeris yang tidak valid, lihat: Memecahkan masalah ephemerides yang tidak valid

Contoh: Mengunggah data ephemeris OEM dari bucket S3

Dimungkinkan juga untuk mengunggah file ephemeris OEM langsung dari bucket S3 dengan menunjuk ke bucket dan kunci objek.AWS Ground Station akan mengambil objek atas nama Anda. Informasi tentang enkripsi data saat istirahat AWS Ground Station dirinci dalam:Enkripsi data saat istirahat untuk AWS Ground Station.

Di bawah ini adalah contoh mengunggah file ephemeris OEM dari bucket S3

```
import boto3
from datetime import datetime, timedelta, timezone
```

```
# Create AWS clients
s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

# Define S3 bucket and key
bucket_name = "ephemeris-bucket"
object_key = "test_data.oem"

# Create sample OEM data in KVN format
oem_data = """CCSDS_OEM_VERS = 2.0

COMMENT Orbit data are consistent with planetary ephemeris DE-430

CREATION_DATE  = 2024-07-22T05:20:59
ORIGINATOR     = Raytheon-JPSS/CGS

META_START
OBJECT_NAME          = J1
OBJECT_ID            = 2017-073A
CENTER_NAME          = Earth
REF_FRAME            = EME2000
TIME_SYSTEM          = UTC
START_TIME           = 2024-07-22T00:00:00.000000
STOP_TIME            = 2024-07-22T00:06:00.000000
INTERPOLATION        = Lagrange
INTERPOLATION_DEGREE = 5
META_STOP

2024-07-22T00:00:00.000000   5.905147360000000e+02  -1.860082793999999e+03
  -6.944807075000000e+03  -5.784245796000000e+00   4.347501391999999e+00
 -1.657256863000000e+00
2024-07-22T00:01:00.000000   2.425572045154201e+02  -1.595860765983339e+03
  -7.030938457373539e+03  -5.810660250794190e+00   4.457103652219009e+00
 -1.212889340333023e+00
2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
  -7.090262617183503e+03  -5.814973972202444e+00   4.549739160042560e+00
 -7.639633689161465e-01
2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
  -7.122556683227951e+03  -5.797176562437553e+00   4.625064829516728e+00
 -3.121687831090774e-01
2024-07-22T00:04:00.000000  -8.015427368657785e+02  -7.709137891269565e+02
  -7.127699477194810e+03  -5.757338007808417e+00   4.682800822515077e+00
 1.407953645161997e-01
```

```
2024-07-22T00:05:00.000000  -1.145240083085062e+03  -4.886583601179489e+02
  -7.105671911254255e+03  -5.695608435738609e+00   4.722731329786999e+00
 5.932259682105052e-01
2024-07-22T00:06:00.000000  -1.484582479061495e+03  -2.045451985605701e+02
  -7.056557069672793e+03  -5.612218005854990e+00   4.744705579872771e+00
 1.043421397392599e+00
"""

# Upload sample OEM data to S3
print(f"Uploading OEM data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name, Key=object_key, Body=oem_data, ContentType="text/plain"
)

print("OEM data uploaded successfully to S3")

# Create OEM ephemeris from S3
print("Creating OEM ephemeris from S3...")

s3_oem_ephemeris = ground_station_client.create_ephemeris(
    name="2024-07-22 S3 OEM Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,
    ephemeris={"oem": {"s3Object": {"bucket": bucket_name, "key": object_key}}},
)

print(f"Created OEM ephemeris with ID: {s3_oem_ephemeris['ephemerisId']}")
```

Di bawah ini adalah contoh data yang dikembalikan dari DescribeEphemeristindakan yang dipanggil untuk ephemeris OEM yang diunggah di blok kode contoh sebelumnya.

```
{
  "creationTime": 1620254718.765,
  "enabled": true,
  "name": "Example Ephemeris",
  "ephemerisId": "fde41049-14f7-413e-bd7b-EXAMPLE02",
  "priority": 2,
  "status": "VALIDATING",
  "suppliedData": {
    "oem": {
```

```
      "sourceS3Object": {
          "bucket": "ephemeris-bucket-for-testing",
          "key": "test_data.oem"
      }
    }
  }
}
```

Berikan data ephemeris elevasi azimuth

> **Important**
>
> Fitur azimuth elevation ephemeris saat ini dalam status Pratinjau dan memerlukan orientasi eksplisit.
> Fungsionalitas ephemeris elevasi Azimuth berada di bawah kontrol akses yang ketat untuk sejumlah kasus penggunaan khusus yang telah ditentukan sebelumnya. Akses secara signifikan lebih ketat daripada kemampuan ephemeris standar yang disediakan pelanggan. Untuk informasi lebih lanjut tentang kasus penggunaan yang disetujui dan proses permintaan akses, silakan buka AWS Dukungan tiket melalui AWS Support Center Console. Tim kami akan memandu Anda melalui proses persetujuan untuk kasus penggunaan khusus.

Gambaran umum

Ephemeris elevasi Azimuth menyediakan cara untuk secara langsung menentukan arah penunjuk antena tanpa memberikan informasi orbital satelit. Alih-alih mengunggah data ephemeris yang menggambarkan orbit satelit, Anda memberikan azimuth dan sudut elevasi yang diberi tag waktu yang memberi tahu antena dengan tepat ke mana harus menunjuk ke seluruh kontak.

AWS Ground Station memperlakukan ephemerides sebagai Data Penggunaan Individual. Jika Anda menggunakan fitur opsional ini, AWS akan menggunakan data ephemeris Anda untuk memberikan dukungan pemecahan masalah.

Pendekatan ini sangat berguna untuk skenario berikut:

- Dukungan operasi awal: Selama Peluncuran dan Fase Orbit Awal (LEOP) ketika data orbital yang tepat tidak tersedia, atau parameter orbital berubah dengan cepat.
- Pola penunjuk khusus: Menerapkan urutan penunjuk khusus untuk pengujian antena atau operasi non-standar.

> **Note**
>
> Saat menggunakan azimuth elevation ephemeris, ARN satelit dapat dihilangkan dari permintaan reservasi kontak. Jika ARN satelit tidak dihilangkan, itu masih akan dimasukkan sebagai bagian dari data kontak, tetapi ephemeris elevasi azimuth akan digunakan untuk penunjuk antena daripada melakukan resolusi prioritas ephemeris. Ephemeris elevasi azimuth dikaitkan dengan stasiun bumi tertentu dan mendefinisikan arah penunjuk antena untuk lokasi itu.

Format data ephemeris elevasi Azimuth

Data ephemeris elevasi Azimuth terdiri dari azimuth dan nilai elevasi yang ditandai dengan waktu yang diatur ke dalam segmen. Setiap segmen berisi serangkaian sudut azimuth dan elevasi yang mencakup rentang waktu tertentu.

Komponen kunci dari data ephemeris elevasi azimuth adalah:

- Ground Station: Stasiun bumi khusus tempat ephemeris elevasi azimuth ini akan digunakan.
- Satuan Sudut: Satuan pengukuran untuk sudut (`DEGREE_ANGLE`atau`RADIAN`).
- Segmen: Satu atau lebih koleksi sudut azimuth dan elevasi yang dibatasi waktu.
- Time-tagged sudut: Nilai azimuth dan elevasi individu dengan stempel waktu terkait.

Setiap segmen membutuhkan:

- Epoch referensi (waktu dasar untuk segmen)
- Rentang waktu yang valid (waktu mulai dan berakhir untuk segmen)
- Setidaknya 5 pasangan yang diberi tag waktu azimuth/elevation

Kendala elevasi Azimuth:

- Azimuth dalam derajat: -180° hingga 360°
- Azimuth dalam radian: -π ke 2π
- Ketinggian dalam derajat: -90° hingga 90°
- Ketinggian dalam radian: -π/2 ke π/2

- Nilai waktu harus dalam urutan menaik dalam setiap segmen
- Segmen tidak boleh tumpang tindih dalam waktu

Untuk informasi selengkapnya, lihat dokumentasi CreateEphemerisAPI dan tipe TimeAzEldata.

Membuat ephemeris elevasi azimuth

Ephemeris elevasi Azimuth dibuat menggunakan aksi CreateEphemerisAPI yang sama, tetapi dengan tipe ephemeris. `azEl` Perbedaan utama dari TLE dan OEM ephemeris adalah:

- Anda harus menentukan `groundStation` parameter
- `satelliteId`Parameter harus dihilangkan dari permintaan
- Pengaturan prioritas tidak berlaku (setiap ephemeris elevasi azimuth khusus untuk stasiun bumi)
- Setiap segmen harus berisi setidaknya 5 azimuth/elevation poin untuk mendukung interpolasi Lagrange orde ke-4
- Batas dan persyaratan tambahan dirinci dalam dokumentasi CreateEphemerisAPI

Penting untuk dicatat bahwa mengunggah ephemeris menyetel ephemeris `VALIDATING` dan memulai alur kerja asinkron yang akan memvalidasi dan menghasilkan kontak potensial dari ephemeris Anda. Ephemeris hanya akan digunakan untuk kontak setelah melewati alur kerja ini dan statusnya menjadi. `ENABLED` Anda harus melakukan polling DescribeEphemerisuntuk status ephemeris atau menggunakan CloudWatch peristiwa untuk melacak perubahan status ephemeris.

Untuk memecahkan masalah ephemeris yang tidak valid, lihat: Memecahkan masalah ephemerides yang tidak valid

Contoh: Buat ephemeris elevasi azimuth melalui API

Contoh berikut menunjukkan cara membuat ephemeris elevasi azimuth menggunakan SDK for AWS Python (Boto3):

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create azimuth elevation ephemeris
```

```
azimuth_elevation_ephemeris = ground_station_client.create_ephemeris(
    name="Azimuth Elevation for Ohio Ground Station",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-03-15T10:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-03-15T10:00:00Z",
                                "endTime": "2024-03-15T10:15:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        }
                    ],
                }
            },
        }
    },
)

print(f"Created ephemeris with ID: {azimuth_elevation_ephemeris['ephemerisId']}")
```

Dalam contoh ini:

- Data elevasi azimuth dikaitkan dengan stasiun bumi “Ohio 1"
- Sudut ditentukan dalam derajat
- Segmen ini mencakup periode 15 menit
- `dt`Nilainya adalah detik atom yang diimbangi dari zaman referensi
- Enam azimuth/elevation pasang disediakan (minimal 5)

Contoh: Unggah data elevasi azimuth dari S3

Untuk kumpulan data yang lebih besar, Anda dapat mengunggah data elevasi azimuth dari bucket S3:

```
import boto3
import json

# Create AWS clients
s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

# Define S3 bucket and key
bucket_name = "azimuth-elevation-bucket"
object_key = "singapore-azimuth-elevation.json"

# Create sample azimuth elevation data
azimuth_elevation_data = {
    "angleUnit": "DEGREE_ANGLE",
    "azElSegmentList": [
        {
            "referenceEpoch": "2024-03-15T10:00:00Z",
            "validTimeRange": {
                "startTime": "2024-03-15T10:00:00Z",
                "endTime": "2024-03-15T10:15:00Z",
            },
            "azElList": [
                {"dt": 0.0, "az": 45.0, "el": 10.0},
                {"dt": 180.0, "az": 50.0, "el": 15.0},
                {"dt": 360.0, "az": 55.0, "el": 20.0},
                {"dt": 540.0, "az": 60.0, "el": 25.0},
                {"dt": 720.0, "az": 65.0, "el": 30.0},
                {"dt": 900.0, "az": 70.0, "el": 35.0},
            ],
        },
        {
            "referenceEpoch": "2024-03-15T10:15:00Z",
            "validTimeRange": {
                "startTime": "2024-03-15T10:15:00Z",
                "endTime": "2024-03-15T10:30:00Z",
            },
            "azElList": [
                {"dt": 0.0, "az": 70.0, "el": 35.0},
```

```
                {"dt": 180.0, "az": 75.0, "el": 40.0},
                {"dt": 360.0, "az": 80.0, "el": 45.0},
                {"dt": 540.0, "az": 85.0, "el": 50.0},
                {"dt": 720.0, "az": 90.0, "el": 55.0},
                {"dt": 900.0, "az": 95.0, "el": 50.0},
            ],
        },
    ],
}

# Upload sample data to S3
print(f"Uploading azimuth elevation data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name,
    Key=object_key,
    Body=json.dumps(azimuth_elevation_data, indent=2),
    ContentType="application/json",
)
print("Sample data uploaded successfully to S3")

# Create azimuth elevation ephemeris from S3
print("Creating azimuth elevation ephemeris from S3...")

s3_azimuth_elevation_ephemeris = ground_station_client.create_ephemeris(
    name="Large Azimuth Elevation Dataset",
    ephemeris={
        "azEl": {
            "groundStation": "Singapore 1",
            "data": {"s3Object": {"bucket": bucket_name, "key": object_key}},
        }
    },
)

print(f"Created ephemeris with ID: {s3_azimuth_elevation_ephemeris['ephemerisId']}")
```

Objek S3 harus berisi struktur JSON dengan data elevasi azimuth dalam format yang sama seperti yang ditunjukkan pada contoh upload langsung.

Memesan kontak dengan ephemeris elevasi azimuth

Saat menggunakan ephemeris elevasi azimuth untuk memesan kontak, prosesnya berbeda dari ephemeris TLE dan OEM:

1. Buat ephemeris elevasi azimuth menggunakan CreateEphemeris
2. Tunggu ephemeris mencapai status ENABLED
3. Pesan kontak menggunakan ReserveContactdengan penggantian pelacakan

Contoh pemesanan kontak dengan ephemeris elevasi azimuth:

```
import boto3
from datetime import datetime
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# First, create an azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")

create_ephemeris_response = ground_station_client.create_ephemeris(
    name="Azimuth Elevation for Contact Reservation",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-03-15T10:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-03-15T10:00:00Z",
                                "endTime": "2024-03-15T10:15:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        }
                    ],
                }
```

```
            },
        }
    },
)

ephemeris_id = create_ephemeris_response["ephemerisId"]
print(f"Created ephemeris with ID: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
print("Waiting for ephemeris to become ENABLED...")

while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# Reserve contact with azimuth elevation ephemeris
print("Reserving contact...")

contact = ground_station_client.reserve_contact(
    # Note: satelliteArn is omitted when using azimuth elevation ephemeris
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-mission-profile",
    groundStation="Ohio 1",
    startTime=datetime(2024, 3, 15, 10, 0, 0),
    endTime=datetime(2024, 3, 15, 10, 15, 0),
    trackingOverrides={"programTrackSettings": {"azEl": {"ephemerisId":
 ephemeris_id}}},
)

print(f"Reserved contact with ID: {contact['contactId']}")
```

> **Note**
>
> `satelliteArn`Parameter dapat dihilangkan saat memesan kontak dengan ephemeris elevasi azimuth. Antena akan mengikuti azimuth dan sudut elevasi yang ditentukan selama kontak.

Daftar kontak yang tersedia

Saat menggunakan ephemeris elevasi azimuth, API memerlukan parameter khusus: ListContacts

- `satelliteArn`Parameter dapat dihilangkan dari permintaan
- Anda harus memberikan `ephemeris` parameter dengan ID ephemeris elevasi azimuth untuk menentukan ephemeris mana yang akan digunakan
- Jendela kontak yang tersedia ditampilkan ketika sudut azimuth dan elevasi yang disediakan berada di atas topeng situs stasiun bumi yang diminta
- Anda harus tetap menyediakan `groundStation` dan `missionProfileArn`

Contoh membuat ephemeris elevasi azimuth dan mencantumkan kontak yang tersedia dengannya:

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Step 1: Create azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Stockholm AzEl Ephemeris",
    ephemeris={
        "azEl": {
            "groundStation": "Stockholm 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-04-01T12:00:00Z",
```

```
                            "validTimeRange": {
                                "startTime": "2024-04-01T12:00:00Z",
                                "endTime": "2024-04-01T12:30:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 30.0, "el": 15.0},
                                {"dt": 360.0, "az": 45.0, "el": 30.0},
                                {"dt": 720.0, "az": 60.0, "el": 45.0},
                                {"dt": 1080.0, "az": 75.0, "el": 35.0},
                                {"dt": 1440.0, "az": 90.0, "el": 20.0},
                                {"dt": 1800.0, "az": 105.0, "el": 10.0},
                            ],
                        }
                    ],
                }
            },
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Step 2: Wait for ephemeris to become ENABLED
print("Waiting for ephemeris to become ENABLED...")
while True:
    describe_response = ground_station_client.describe_ephemeris(
        ephemerisId=ephemeris_id
    )
    status = describe_response["status"]

    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        # Check for validation errors
        if "invalidReason" in describe_response:
            print(f"Ephemeris validation failed: {describe_response['invalidReason']}")
        raise RuntimeError(f"Ephemeris failed with status: {status}")

    print(f"Current status: {status}, waiting...")
    time.sleep(5)

# Step 3: List available contacts using the azimuth elevation ephemeris
```

```
print("Listing available contacts with azimuth elevation ephemeris...")

# Convert epoch timestamps to datetime objects
start_time = datetime.fromtimestamp(1760710513, tz=timezone.utc)
end_time = datetime.fromtimestamp(1760883313, tz=timezone.utc)

contacts_response = ground_station_client.list_contacts(
    startTime=start_time,
    endTime=end_time,
    groundStation="Stockholm 1",
    statusList=["AVAILABLE"],
    ephemeris={"azEl": {"id": ephemeris_id}},
    # satelliteArn is optional
    satelliteArn="arn:aws:groundstation::111122223333:satellite/a88611b0-f755-404e-
b60d-57d8aEXAMPLE",
    missionProfileArn="arn:aws:groundstation:eu-north-1:111122223333:mission-
profile/966b72f6-6d82-4e7e-b072-f8240EXAMPLE",
)

# Process the results
if contacts_response["contactList"]:
    print(f"Found {len(contacts_response['contactList'])} available contacts:")
    for contact in contacts_response["contactList"]:
        print(f"  - Contact from {contact['startTime']} to {contact['endTime']}")
        print(
            f"    Max elevation: {contact.get('maximumElevation', {}).get('value', 'N/
A')}°"
        )
else:
    print("No available contacts found for the specified azimuth elevation ephemeris")
```

> **Note**
>
> `ephemeris`Parameter dengan ID elevasi azimuth harus disediakan saat mencantumkan kontak untuk menentukan ephemeris elevasi azimuth mana yang harus digunakan untuk menentukan jendela kontak. Jika disertakan, itu akan dikaitkan dengan data kontak, tetapi ephemeris elevasi azimuth akan digunakan untuk penunjuk antena daripada melakukan resolusi prioritas ephemeris. `satelliteArn`

Cadangan kontak dengan ephemeris khusus

Gambaran umum

Saat menggunakan ephemeris khusus (TLE, OEM, atau elevasi azimuth), Anda dapat memesan kontak menggunakan API. ReserveContact Bagian ini menjelaskan dua alur kerja umum untuk pemesanan kontak dan pertimbangan penting untuk memastikan penjadwalan kontak berhasil.

AWS Ground Station antena adalah sumber daya bersama di antara banyak pelanggan. Ini berarti bahwa meskipun jendela kontak muncul tersedia saat Anda mencantumkan kontak, pelanggan lain mungkin memesannya sebelum Anda melakukannya. Oleh karena itu, penting untuk memverifikasi bahwa kontak Anda mencapai `SCHEDULED` negara setelah reservasi dan menerapkan pemantauan yang tepat untuk perubahan status kontak.

> **Important**
>
> Untuk ephemeris elevasi azimuth, `satelliteArn` parameter dapat dihilangkan dari `ReserveContact` permintaan, dan Anda harus memberikan ID ephemeris. `trackingOverrides` Untuk ephemeris TLE dan OEM, Anda masih perlu menyediakan. `satelliteArn`

Alur kerja reservasi kontak

Ada dua alur kerja utama untuk memesan kontak dengan ephemeris khusus:

1. List-then-reserve alur kerja: Daftar pertama jendela kontak yang tersedia menggunakan ListContacts, lalu pilih dan pesan jendela tertentu. Pendekatan ini berguna ketika Anda ingin melihat semua peluang yang tersedia sebelum membuat pilihan.
2. Alur kerja reservasi langsung: Langsung pesan kontak untuk jendela waktu tertentu tanpa terlebih dahulu mencantumkan kontak yang tersedia. Pendekatan ini berguna ketika Anda sudah mengetahui waktu kontak yang Anda inginkan atau bekerja dengan jadwal yang telah ditentukan.

Kedua alur kerja valid dan pilihannya tergantung pada kebutuhan operasional Anda. Bagian berikut memberikan contoh dari setiap pendekatan.

Alur kerja 1: Daftar kontak yang tersedia lalu pesan

Alur kerja ini pertama-tama menanyakan jendela kontak yang tersedia, lalu mencadangkan jendela tertentu. Ini berguna ketika Anda ingin melihat semua peluang yang tersedia sebelum membuat pilihan.

Contoh: Daftar dan cadangan dengan ephemeris elevasi azimuth

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="AzEl Ephemeris for Contact",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-03-15T10:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-03-15T10:00:00Z",
                                "endTime": "2024-03-15T10:15:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        }
                    ],
                }
```

```
            },
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# List available contacts
print("Listing available contacts...")
contacts = ground_station_client.list_contacts(
    # Note: satelliteArn is omitted for azimuth elevation ephemeris
    groundStation="Ohio 1",
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-profile",
    startTime=datetime(2024, 3, 15, 10, 0, 0, tzinfo=timezone.utc),
    endTime=datetime(2024, 3, 15, 10, 15, 0, tzinfo=timezone.utc),
    statusList=["AVAILABLE"],
    ephemeris={"azEl": {"id": ephemeris_id}},
)

if contacts["contactList"]:
    # Reserve the first available contact
    contact = contacts["contactList"][0]
    print(f"Reserving contact from {contact['startTime']} to {contact['endTime']}...")

    reservation = ground_station_client.reserve_contact(
        # Note: satelliteArn is omitted when using azimuth elevation ephemeris
        missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-
profile/example-profile",
        groundStation="Ohio 1",
        startTime=contact["startTime"],
```

```
        endTime=contact["endTime"],
        trackingOverrides={
            "programTrackSettings": {"azEl": {"ephemerisId": ephemeris_id}}
        },
    )

    print(f"Reserved contact: {reservation['contactId']}")
else:
    print("No available contacts found")
```

Contoh: Daftar dan pesan dengan TLE ephemeris

```
import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

satellite_id = "12345678-1234-1234-1234-123456789012"
satellite_arn = f"arn:aws:groundstation::111122223333:satellite/{satellite_id}"

# Create TLE ephemeris
print("Creating TLE ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="TLE Ephemeris for Contact",
    satelliteId=satellite_id,
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=7),
    priority=1,  # Higher priority than default ephemeris
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   24075.54719794  .00000075  00000-0
 26688-4 0  9997",
                    "tleLine2": "2 25994  98.2007  30.6589 0001234  89.2782  18.9934
 14.57114995111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
```

```
            ]
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# List available contacts
print("Listing available contacts...")
start_time = datetime.now(timezone.utc) + timedelta(hours=1)
end_time = start_time + timedelta(days=1)

contacts = ground_station_client.list_contacts(
    satelliteArn=satellite_arn,  # Required for TLE/OEM ephemeris
    groundStation="Hawaii 1",
    missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-profile/
example-profile",
    startTime=start_time,
    endTime=end_time,
    statusList=["AVAILABLE"],
)

if contacts["contactList"]:
    # Reserve the first available contact
    contact = contacts["contactList"][0]
    print(f"Reserving contact from {contact['startTime']} to {contact['endTime']}...")

    reservation = ground_station_client.reserve_contact(
        satelliteArn=satellite_arn,  # Required for TLE/OEM ephemeris
        missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-
profile/example-profile",
```

```
        groundStation="Hawaii 1",
        startTime=contact["startTime"],
        endTime=contact["endTime"],
        # Note: trackingOverrides is optional for TLE/OEM
        # The system will use the highest priority ephemeris automatically
    )

    print(f"Reserved contact: {reservation['contactId']}")
else:
    print("No available contacts found")
```

Alur kerja 2: Reservasi kontak langsung

Alur kerja ini secara langsung mencadangkan kontak tanpa mencantumkan jendela yang tersedia terlebih dahulu. Pendekatan ini berguna ketika Anda sudah mengetahui waktu kontak yang Anda inginkan atau menerapkan penjadwalan otomatis.

Contoh: Reservasi langsung dengan ephemeris elevasi azimuth

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Define contact window
contact_start = datetime(2024, 3, 20, 14, 0, 0, tzinfo=timezone.utc)
contact_end = datetime(2024, 3, 20, 14, 15, 0, tzinfo=timezone.utc)

# Create azimuth elevation ephemeris for the specific contact time
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact AzEl Ephemeris",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": contact_start.isoformat(),
```

```
                            "validTimeRange": {
                                "startTime": contact_start.isoformat(),
                                "endTime": contact_end.isoformat(),
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        }
                    ],
                }
            },
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# Directly reserve the contact
print(f"Reserving contact from {contact_start} to {contact_end}...")

reservation = ground_station_client.reserve_contact(
    # Note: satelliteArn is omitted for azimuth elevation
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-profile",
    groundStation="Ohio 1",
    startTime=contact_start,
```

```
    endTime=contact_end,
    trackingOverrides={"programTrackSettings": {"azEl": {"ephemerisId":
 ephemeris_id}}},
)

print(f"Reserved contact: {reservation['contactId']}")
```

Contoh: Reservasi langsung dengan TLE ephemeris

```
import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

satellite_id = "12345678-1234-1234-1234-123456789012"
satellite_arn = f"arn:aws:groundstation::111122223333:satellite/{satellite_id}"

# Define contact window (based on predicted pass)
contact_start = datetime(2024, 3, 21, 10, 30, 0, tzinfo=timezone.utc)
contact_end = datetime(2024, 3, 21, 10, 42, 0, tzinfo=timezone.utc)

# Create TLE ephemeris
print("Creating TLE ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact TLE Ephemeris",
    satelliteId=satellite_id,
    enabled=True,
    expirationTime=contact_end + timedelta(days=1),
    priority=1,
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   24080.50000000  .00000075  00000-0
 26688-4 0  9999",
                    "tleLine2": "2 25994  98.2007  35.6589 0001234  89.2782  18.9934
 14.57114995112000",
                    "validTimeRange": {
                        "startTime": (contact_start - timedelta(hours=1)).isoformat(),
                        "endTime": (contact_end + timedelta(hours=1)).isoformat(),
                    },
```

```
                }
            ]
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# Directly reserve the contact
print(f"Reserving contact from {contact_start} to {contact_end}...")

reservation = ground_station_client.reserve_contact(
    satelliteArn=satellite_arn,  # Required for TLE ephemeris
    missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-profile/
example-profile",
    groundStation="Hawaii 1",
    startTime=contact_start,
    endTime=contact_end,
    # Note: trackingOverrides is optional for TLE
    # The system will use the highest priority ephemeris automatically
)

print(f"Reserved contact: {reservation['contactId']}")
```

Memantau perubahan status kontak

Setelah memesan kontak, penting untuk memantau statusnya untuk memastikannya berhasil bertransisi ke SCHEDULED dan diberi tahu tentang masalah apa pun.AWS Ground Station memancarkan peristiwa ke Amazon EventBridge untuk semua perubahan status kontak.

Status kontak mengikuti siklus hidup ini:

- `SCHEDULING`- Kontak sedang diproses untuk penjadwalan
- `SCHEDULED`- Kontak berhasil dijadwalkan dan akan dieksekusi
- `FAILED_TO_SCHEDULE`- Kontak tidak dapat dijadwalkan (status terminal)

Untuk informasi selengkapnya tentang status kontak dan siklus hidup, lihat. Memahami siklus hidup kontak

Menerapkan pemantauan status kontak dengan EventBridge

Untuk memantau perubahan status kontak secara real-time, Anda dapat mengatur EventBridge aturan Amazon yang memicu fungsi Lambda setiap kali kontak Ground Station berubah status. Pendekatan ini lebih efisien dan terukur daripada polling status kontak.

Langkah-langkah implementasi

1. Buat fungsi Lambda untuk memproses peristiwa perubahan status kontak
2. Buat EventBridge aturan yang cocok dengan peristiwa perubahan status kontak Ground Station
3. Tambahkan fungsi Lambda sebagai target aturan

Contoh penangan fungsi Lambda

Untuk contoh lengkap fungsi Lambda yang memproses peristiwa perubahan status kontak, lihat `GroundStationCloudWatchEventHandlerLambda` sumber daya dalam templat. `AquaSnppJpssTerraDigIF.yml`CloudFormation Template ini tersedia di bucket Amazon S3 orientasi AWS Ground Station pelanggan. Untuk petunjuk tentang mengakses template ini, lihat Menyatukannya bagian dari contoh titik akhir aliran data.

EventBridge konfigurasi aturan

EventBridge Aturan harus menggunakan pola peristiwa berikut untuk mencocokkan semua perubahan status kontak Ground Station:

```
{
  "source": ["aws.groundstation"],
  "detail-type": ["Ground Station Contact State Change"]
}
```

Untuk memfilter hanya untuk status tertentu (misalnya, kegagalan), Anda dapat menambahkan filter detail:

```
{
  "source": ["aws.groundstation"],
  "detail-type": ["Ground Station Contact State Change"],
  "detail": {
    "contactStatus": [
      "FAILED_TO_SCHEDULE",
      "FAILED",
      "AWS_FAILED",
      "AWS_CANCELLED"
    ]
  }
}
```

Untuk petunjuk mendetail tentang cara membuat EventBridge aturan dengan target Lambda, lihat Membuat aturan yang bereaksi terhadap peristiwa di EventBridge Panduan Pengguna Amazon.

Menyiapkan EventBridge aturan untuk otomatisasi

Anda dapat membuat EventBridge aturan untuk merespons perubahan status kontak secara otomatis. Contoh:

- Kirim pemberitahuan saat kontak gagal menjadwalkan
- Memicu fungsi Lambda untuk menyiapkan sumber daya saat kontak masuk `PREPASS`
- Log penyelesaian kontak untuk tujuan audit

Untuk informasi rinci tentang pengaturan EventBridge aturan untuk AWS Ground Station acara, lihatOtomatisasi AWS Ground Station dengan Acara.

Praktik terbaik dan pertimbangan

Menangani konflik penjadwalan

Karena AWS Ground Station antena adalah sumber daya bersama, jendela kontak yang muncul tersedia `ListContacts` mungkin dipesan oleh pelanggan lain sebelum Anda dapat memesannya. Untuk menangani ini:

1. Selalu periksa status kontak setelah reservasi

2. Terapkan logika coba lagi dengan jendela waktu alternatif
3. Pertimbangkan untuk memesan kontak jauh sebelumnya jika memungkinkan
4. Gunakan EventBridge acara untuk memantau `FAILED_TO_SCHEDULE` negara

Waktu validasi Ephemeris

Ingat bahwa ephemeris harus dalam `ENABLED` keadaan sebelum Anda dapat menggunakannya untuk memesan kontak. Proses validasi biasanya memakan waktu beberapa detik hingga beberapa menit tergantung pada jenis dan ukuran ephemeris. Selalu verifikasi status ephemeris sebelum mencoba memesan kontak.

Pertimbangan waktu kontak

Saat menggunakan ephemeris khusus:

- Pastikan ephemeris Anda mencakup seluruh durasi kontak
- Untuk ephemeris elevasi azimuth, verifikasi bahwa sudut menjaga antena di atas situs menutupi seluruh kontak
- Pertimbangkan waktu kedaluwarsa ephemeris saat menjadwalkan kontak masa depan

Perbedaan API menurut tipe ephemeris

`ReserveContact`API berperilaku berbeda tergantung pada tipe ephemeris:

Jenis Ephemeris	SatellitEarn Diperlukan	TrackingOverrides Diperlukan
TLE	Ya	Tidak (opsional)
OEM	Ya	Tidak (opsional)
Ketinggian Azimuth	Tidak (opsional)	Ya

Memahami ephemeris mana yang digunakan

Ephemerides memiliki prioritas, waktu kedaluwarsa, dan flag yang diaktifkan. Bersama-sama, ini menentukan ephemeris mana yang digunakan untuk melacak selama kontak.

Ephemerides TLE dan OEM

Untuk ephemerides OEM dan TLE, hanya satu ephemeris yang dapat aktif untuk setiap satelit. Ephemeris yang akan digunakan adalah ephemeris dengan prioritas tertinggi yang waktu kedaluwarsa di masa depan. Nilai prioritas yang lebih besar menunjukkan prioritas yang lebih tinggi. Waktu kontak yang tersedia yang dikembalikan oleh ListContactsdidasarkan pada ephemeris ini. Jika beberapa `ENABLED` ephemerides memiliki prioritas yang sama, ephemeris yang terbaru dibuat atau diperbarui akan digunakan.

> **Note**
>
> AWS Ground Stationmemiliki kuota layanan pada jumlah ephemerides yang `ENABLED` disediakan pelanggan per satelit (lihat: Service Quotas). Untuk mengunggah data ephemeris setelah mencapai kuota ini, hapus (menggunakan DeleteEphemeris) atau nonaktifkan (menggunakan UpdateEphemeris) ephemerides yang disediakan pelanggan yang paling rendahpriority/earliest .

Jika tidak ada ephemeris yang dibuat, atau jika tidak ada ephemerides yang `ENABLED` berstatus,AWS Ground Station akan menggunakan ephemeris default untuk satelit (dari), jika tersedia. Space-Track Ephemeris default ini memiliki prioritas 0.

Ephemerides elevasi Azimuth

Ephemerides elevasi Azimuth bekerja secara berbeda dari ephemerides OEM dan TLE. Setiap ephemeris elevasi azimuth dikaitkan dengan stasiun bumi tertentu dan tidak memiliki prioritas. Saat Anda memesan kontak dengan ephemeris elevasi azimuth, Anda secara eksplisit menentukan ephemeris elevasi azimuth mana yang akan digunakan melalui parameter. `trackingOverrides`

Perbedaan utama untuk ephemerides elevasi azimuth:

- Tidak ada sistem prioritas - Anda secara eksplisit memilih ephemeris untuk setiap kontak
- Stasiun darat spesifik - setiap ephemeris dikaitkan dengan stasiun bumi tertentu
- Tidak ada fallback otomatis - jika ephemeris yang ditentukan tidak tersedia, kontak akan gagal

> **Note**
>
> Ephemerides elevasi Azimuth tidak bersaing dengan ephemerides OEM dan TLE. Mereka dipilih secara eksplisit saat memesan kontak dan hanya digunakan saat pelacakan penggantian ditentukan.

Pengaruh ephemerides baru pada kontak yang dijadwalkan sebelumnya

Gunakan DescribeContact API untuk melihat efek ephemerides baru pada kontak yang dijadwalkan sebelumnya dengan mengembalikan waktu visibilitas aktif.

Untuk ephemerides OEM dan TLE, kontak yang dijadwalkan sebelum mengunggah ephemeris baru akan mempertahankan waktu kontak yang dijadwalkan semula, sedangkan pelacakan antena akan menggunakan ephemeris aktif. Jika posisi pesawat ruang angkasa, berdasarkan ephemeris aktif, sangat berbeda dari ephemeris sebelumnya, ini dapat mengakibatkan berkurangnya waktu kontak satelit dengan antena karena pesawat ruang angkasa yang beroperasi di luar topeng situs. transmit/receive Oleh karena itu, kami menyarankan Anda membatalkan dan menjadwal ulang kontak future Anda setelah Anda mengunggah ephemeris baru yang sangat berbeda dari ephemeris sebelumnya.

Dengan DescribeContact API, Anda dapat menentukan bagian dari kontak future Anda yang tidak dapat digunakan karena pesawat ruang angkasa yang beroperasi di luar topeng transmit/receive situs dengan membandingkan kontak terjadwal Anda `startTime` dan `endTime` dengan yang dikembalikan `visibilityStartTime` dan. `visibilityEndTime` Jika Anda memilih untuk membatalkan dan menjadwal ulang kontak masa depan Anda, rentang waktu kontak tidak boleh berada di luar rentang waktu visibilitas lebih dari 30 detik. Kontak yang dibatalkan dapat dikenakan biaya jika dibatalkan terlalu dekat dengan waktu kontak. Untuk informasi selengkapnya tentang kontak yang dibatalkan, lihat: FAQ Ground Station.

Untuk ephemerides elevasi azimuth, kontak terjadwal akan menggunakan ephemeris spesifik yang dipilih saat kontak dicadangkan. Jika Anda perlu memperbarui data elevasi azimuth untuk kontak terjadwal, Anda dapat membatalkan dan menjadwal ulang kontak dengan ephemeris baru.

Dapatkan ephemeris saat ini untuk satelit

Ephemeris saat ini digunakan oleh AWS Ground Station untuk satelit tertentu dapat diambil dengan memanggil atau tindakan. GetSatelliteListSatellites Kedua metode ini akan mengembalikan metadata

untuk ephemeris yang saat ini digunakan. Metadata ephemeris ini berbeda untuk ephemerides khusus yang diunggah ke dan untuk ephemerides default.AWS Ground Station

> **Note**
>
> Ephemerides elevasi Azimuth tidak terkait dengan satelit dan oleh karena itu tidak dikembalikan oleh atau. GetSatelliteListSatellites Untuk mengambil informasi tentang ephemerides elevasi azimuth, gunakan DescribeEphemerisAPI dengan ID ephemeris tertentu, atau gunakan untuk melihat semua ephemerides yang tersedia untuk akun Anda. ListEphemerides

Ephemerides default hanya akan menyertakan `source` dan bidang. `epoch` `epoch`Ini adalah zaman dari set elemen dua baris yang ditarik dari Space-Track, dan saat ini sedang digunakan untuk menghitung lintasan satelit.

Ephemeris khusus akan memiliki `source` nilai `CUSTOMER_PROVIDED` dan akan menyertakan pengidentifikasi unik di lapangan. `ephemerisId` Pengidentifikasi unik ini dapat digunakan untuk menanyakan ephemeris melalui tindakan. DescribeEphemeris `name`Bidang opsional akan dikembalikan jika ephemeris diberi nama saat diunggah AWS Ground Station melalui tindakan. CreateEphemeris

Penting untuk dicatat bahwa ephemerides diperbarui secara dinamis AWS Ground Station sehingga data yang dikembalikan hanyalah snapshot dari ephemeris yang digunakan pada saat panggilan ke API.

Contoh GetSatellite kembali untuk satelit menggunakan ephemeris default

```
{
    "satelliteId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
    "satelliteArn": "arn:aws:groundstation::111122223333:satellite/e1cfe0c7-67f9-4d98-
bad2-EXAMPLE",
    "noradSatelliteID": 25994,
    "groundStations": [
        "Ohio 1",
        "Oregon 1"
    ],
    "currentEphemeris": {
        "source": "SPACE_TRACK",
```

```
        "epoch": 1528245583.619
    }
}
```

Contoh GetSatellite untuk satelit menggunakan ephemeris khusus

```
{
      "satelliteId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
      "satelliteArn": "arn:aws:groundstation::111122223333:satellite/
e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
      "noradSatelliteID": 25994,
      "groundStations": [
            "Ohio 1",
            "Oregon 1"
      ],
      "currentEphemeris": {
          "source": "CUSTOMER_PROVIDED",
          "ephemerisId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
          "name": "My Ephemeris"
      }
  }
```

Daftar ephemerides elevasi azimuth

Karena ephemerides elevasi azimuth tidak terkait dengan satelit, Anda perlu menggunakan API yang berbeda untuk menemukan dan mengambil informasi tentang mereka:

1. Gunakan ListEphemeridesuntuk mencantumkan semua ephemerides di akun Anda, termasuk ephemerides elevasi azimuth. Anda dapat memfilter berdasarkan status dan jenis ephemeris.
2. Gunakan DescribeEphemerisdengan ID ephemeris tertentu untuk mendapatkan informasi rinci tentang ephemeris elevasi azimuth.
3. Gunakan DescribeContactdengan ID kontak tertentu untuk mendapatkan informasi terperinci tentang ephemeris yang digunakan untuk kontak tersebut.

Contoh ListEphemeridesrespons termasuk ephemeris elevasi azimuth:

```
{
    "ephemerides": [
        {
```

```
            "ephemerisId": "abc12345-6789-def0-1234-5678EXAMPLE",
            "ephemerisType": "AZ_EL",
            "name": "Azimuth Elevation for Ohio Ground Station",
            "status": "ENABLED",
            "creationTime": 1620254718.765
        },
        {
            "ephemerisId": "def45678-9012-abc3-4567-8901EXAMPLE",
            "ephemerisType": "TLE",
            "name": "TLE for Satellite 12345",
            "status": "ENABLED",
            "creationTime": 1620254700.123
        }
    ]
}
```

> **Note**
>
> ListEphemeridesSebagai tanggapan, ephemerides elevasi azimuth akan memiliki `groundStation` bidang alih-alih `satelliteId` bidang, membuatnya mudah diidentifikasi.

Kembalikan ke data ephemeris default

Saat Anda mengunggah data ephemeris khusus, itu akan mengganti penggunaan AWS Ground Station ephemerides default untuk satelit tertentu.AWS Ground Station tidak menggunakan ephemeris default lagi sampai saat ini tidak ada ephemerides yang disediakan pelanggan yang belum kedaluwarsa yang saat ini tersedia untuk digunakan.AWS Ground Station juga tidak mencantumkan kontak melewati waktu kedaluwarsa ephemeris yang disediakan pelanggan saat ini, bahkan jika ada ephemeris default yang tersedia melewati waktu kedaluwarsa tersebut.

> **Note**
>
> Ephemerides elevasi Azimuth tidak memiliki nilai default dan tidak mengesampingkan ephemerides satelit. Mereka dipilih secara eksplisit saat memesan kontak menggunakan parameter. `trackingOverrides` Jika Anda tidak lagi ingin menggunakan ephemeris elevasi azimuth, cukup pesan kontak tanpa menentukan penggantian pelacakan, dan sistem akan menggunakan ephemeris satelit aktif sebagai gantinya.

Mengembalikan ephemerides TLE dan OEM

Untuk kembali ke Space-Trackephemerides default untuk satelit, Anda perlu melakukan salah satu hal berikut:

- Hapus (menggunakan DeleteEphemeris) atau menonaktifkan (menggunakan UpdateEphemeris) semua ephemerides yang disediakan pelanggan yang diaktifkan. Anda dapat membuat daftar ephemerides yang disediakan pelanggan untuk menggunakan satelit. ListEphemerides
- Tunggu semua ephemerides yang disediakan pelanggan yang ada kedaluwarsa.

Anda dapat mengonfirmasi bahwa ephemeris default sedang digunakan dengan memanggil GetSatellitedan memverifikasi bahwa ephemeris saat ini untuk satelit adalah. `source` `SPACE_TRACK` Lihat Data ephemeris standar untuk informasi lebih lanjut tentang ephemerides default.

Mengelola ephemerides elevasi azimuth

Karena ephemerides elevasi azimuth dipilih secara eksplisit untuk setiap kontak dan tidak terkait dengan satelit, tidak ada konsep “mengembalikan” ke default. Sebagai gantinya, Anda dapat mengelola ephemerides elevasi azimuth sebagai berikut:

- Untuk berhenti menggunakan ephemeris elevasi azimuth: Cukup pesan kontak baru tanpa menentukan dan menentukan a. `trackingOverrides` `satelliteArn` Kontak akan menggunakan ephemeris aktif untuk satelit yang ditentukan sebagai gantinya.
- Untuk menghapus ephemerides elevasi azimuth yang tidak digunakan: Gunakan DeleteEphemerisuntuk menghapus ephemerides elevasi azimuth yang tidak lagi diperlukan. Perhatikan bahwa Anda tidak dapat menghapus ephemeris yang saat ini sedang digunakan oleh kontak terjadwal.

Untuk mencantumkan semua ephemerides elevasi azimuth di akun Anda, gunakan. ListEphemerides Ephemerides elevasi Azimuth dapat diidentifikasi oleh `ephemerisType` lapangan, atau dengan adanya medan, bukan `groundStation` bidang dalam respons. `satelliteId`

Bekerja dengan aliran data

AWS Ground Station menggunakan hubungan simpul dan tepi untuk membangun aliran data untuk memungkinkan pemrosesan aliran data Anda. Setiap node diwakili oleh konfigurasi yang menjelaskan pemrosesan yang diharapkan. Untuk mengilustrasikan konsep ini, pertimbangkan aliran data ke a. `antenna-downlink` `s3-recording` `antenna-downlink`Node mewakili transformasi analog ke digital dari spektrum frekuensi radio per parameter yang ditentukan pada konfigurasi. `s3-recording`Ini mewakili node komputasi yang akan menerima data masuk dan menyimpannya di bucket S3 Anda. Aliran data yang dihasilkan adalah pengiriman data asinkron dari data RF digital ke bucket S3 berdasarkan spesifikasi Anda.

Dalam profil misi Anda, Anda dapat membuat banyak aliran data untuk memenuhi kebutuhan Anda. Bagian berikut menjelaskan cara menyiapkan sumber daya AWS Anda yang lain untuk digunakan AWS Ground Station dan menawarkan rekomendasi untuk membuat aliran data. Untuk informasi rinci tentang bagaimana setiap node berperilaku, termasuk jika dianggap sebagai sumber atau node tujuan, silakan lihatGunakan AWS Ground Station Konfigurasi.

Topik

AWS Ground Station antarmuka bidang data

Struktur data yang dihasilkan dari aliran data yang Anda pilih tergantung pada sumber aliran data. Rincian format ini diberikan kepada Anda selama orientasi satelit Anda. Berikut ini merangkum format yang digunakan untuk setiap jenis aliran data.

- antena-downlink
  - (Bandwidth less-than-or-equal -to 40MHz) data dikirimkan sebagai paket Data Sinyal/Format IP VITA-49.
  - (Bandwidth lebih besar dari 40MHz) data dikirimkan sebagai paket AWS Ground Station Kelas 2.
- antenna-downlink-demod-decode

  - Data dikirimkan sebagai paket Format Demodulated/Decoded Data/IP.
- antena-uplink
  - Data harus dikirimkan sebagai paket VITA-49 Signal Data/IP Format.
- antenna-uplink-echo
  - Data dikirimkan sebagai paket VITA-49 Signal Data/IP Format.

Gunakan pengiriman data lintas wilayah

Fitur pengiriman data AWS Ground Station lintas wilayah memberi Anda fleksibilitas untuk mengirim data Anda dari antena ke AWS Wilayah mana pun yang AWS Ground Station didukung. Ini berarti Anda dapat memelihara infrastruktur Anda di satu Wilayah AWS dan menjadwalkan kontak pada siapa pun yang AWS Ground Station Lokasi Anda gunakan.

Saat menerima data kontak Anda di Amazon S3 Bucket, AWS Ground Station akan mengelola semua aspek pengiriman untuk Anda.

Untuk menggunakan pengiriman data lintas wilayah ke EC2 instans Amazon (menggunakan AWS Ground Station Agen atau titik akhir aliran data), titik akhir rendah data harus dibuat di Wilayah AWS Anda saat ini dan Anda harus menentukan wilayah yang sama. dataflow-endpoint-config AWS Ground Station akan mengelola pengiriman data lintas wilayah untuk Anda.

Siapkan dan konfigurasikan Amazon S3

Anda dapat menggunakan bucket Amazon S3 untuk menerima sinyal downlink Anda menggunakan. AWS Ground Station Untuk membuat konfigurasi s3-recording-tujuan, Anda harus dapat menentukan bucket Amazon S3 dan peran IAM yang mengizinkan untuk menulis file ke bucket. AWS Ground Station

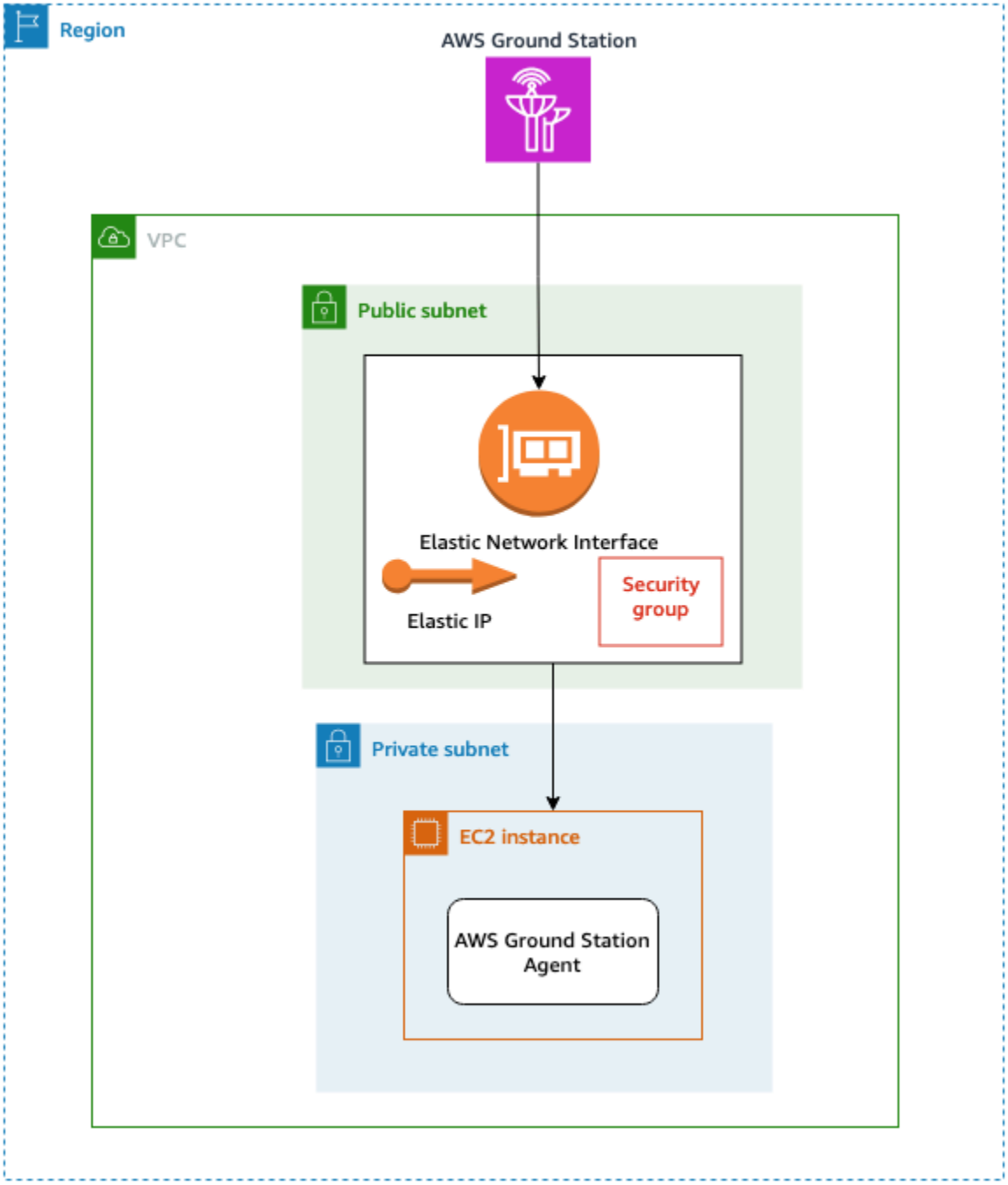
Lihat Config Perekaman Amazon S3 batasan pada bucket Amazon S3, peran IAM, atau AWS Ground Station pembuatan konfigurasi.

Siapkan dan konfigurasikan Amazon VPC

Panduan lengkap untuk menyiapkan VPC berada di luar cakupan panduan ini. Untuk pemahaman mendalam, silakan merujuk ke Panduan Pengguna Amazon VPC.

Di bagian ini, dijelaskan bagaimana Amazon EC2 dan titik akhir aliran data Anda mungkin ada dalam VPC. AWS Ground Station tidak mendukung beberapa titik pengiriman untuk aliran data tertentu - diharapkan setiap aliran data berakhir ke satu penerima. EC2 Seperti yang kami harapkan satu EC2 penerima, konfigurasinya tidak berlebihan Multi-AZ. Untuk contoh lengkap yang akan menggunakan VPC Anda, silakan lihat. Contoh konfigurasi profil misi

Konfigurasi VPC dengan Agen AWS Ground Station



Data satelit Anda diberikan ke instance AWS Ground Station Agen yang dekat dengan antena. AWS Ground Station Agen akan melakukan stripe dan kemudian mengenkripsi data Anda menggunakan AWS KMS kunci yang Anda berikan. Setiap strip dikirim ke Amazon EC2 Elastic IP (EIP) Anda dari antena sumber di seluruh tulang punggung AWS Network. Data tiba di EC2 instans Anda melalui Amazon EC2 Elastic Network Interface (ENI) yang terpasang. Setelah di EC2 instans Anda, AWS Ground Station Agen yang diinstal akan mendekripsi data Anda dan melakukan koreksi kesalahan ke depan (FEC) untuk memulihkan data yang dijatuhkan, lalu meneruskannya ke IP dan port yang Anda tentukan dalam pengaturan Anda.

Daftar di bawah ini menyebutkan pertimbangan pengaturan unik saat menyiapkan VPC Anda AWS Ground Station untuk pengiriman Agen.

Grup Keamanan - Disarankan Anda membuat grup keamanan yang didedikasikan hanya untuk AWS Ground Station lalu lintas. Grup keamanan ini harus mengizinkan lalu lintas masuknya UDP pada rentang port yang sama yang Anda tentukan di Grup Titik Akhir Dataflow Anda. AWS Ground Station mempertahankan daftar awalan yang dikelola AWS untuk membatasi izin Anda hanya ke alamat IP. AWS Ground Station Lihat Daftar Awalan Terkelola AWS untuk detail tentang cara mengganti PrefixListIdwilayah penerapan Anda.

Elastic Network Interface (ENI) - Anda harus mengaitkan grup keamanan di atas dengan ENI ini dan menempatkannya di subnet publik Anda.

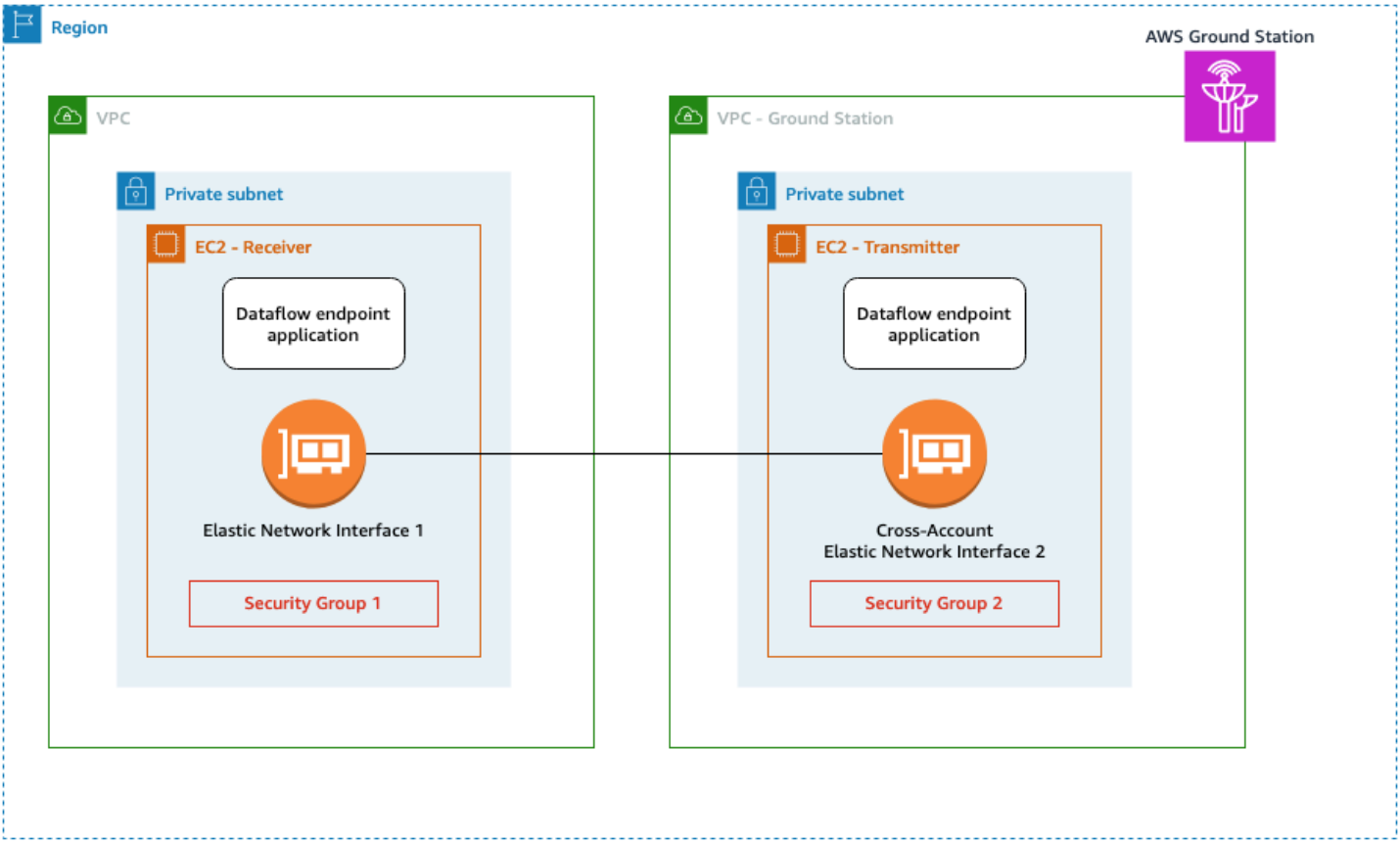
> **Note**
>
> Kuota default untuk jumlah grup keamanan yang dilampirkan per ENI adalah 5. Ini adalah batas yang dapat disesuaikan hingga 16, lihat Kuota VPC Amazon.

CloudFormation Template berikut menunjukkan cara membuat infrastruktur yang dijelaskan di bagian ini.

```
ReceiveInstanceEIP:
  Type: AWS::EC2::EIP
  Properties:
    Domain: 'vpc'

InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
```

```
    VpcId:YourVpcId
    SecurityGroupIngress:
      # Add additional items here.
      - IpProtocol: udp
        FromPort: your-port-start-range
        ToPort: your-port-end-range
        PrefixListIds:
          - PrefixListId: com.amazonaws.global.groundstation
        Description: "Allow AWS Ground Station Downlink ingress."

InstanceNetworkInterface:
  Type: AWS::EC2::NetworkInterface
  Properties:
    Description: ENI for AWS Ground Station to connect to.
    GroupSet:
      - !Ref InstanceSecurityGroup
    SubnetId: A Public Subnet

ReceiveInstanceEIPAllocation:
  Type: AWS::EC2::EIPAssociation
  Properties:
    AllocationId:
      Fn::GetAtt: [ ReceiveInstanceEIP, AllocationId ]
    NetworkInterfaceId:
      Ref: InstanceNetworkInterface
```

Konfigurasi VPC dengan titik akhir aliran data



Data satelit Anda disediakan ke instance aplikasi titik akhir aliran data yang dekat dengan antena. Data tersebut kemudian dikirim melalui lintas akun Amazon EC2 Elastic Network Interface (ENI) dari VPC yang dimiliki oleh. AWS Ground Station Data kemudian tiba di EC2 instans Anda melalui ENI yang dilampirkan ke EC2 instans Amazon Anda. Aplikasi endpoint dataflow yang diinstal kemudian akan meneruskannya ke IP dan port yang Anda tentukan dalam pengaturan Anda. Kebalikan dari aliran ini terjadi untuk koneksi uplink.

Daftar di bawah ini menyebutkan pertimbangan penyiapan unik saat menyiapkan VPC Anda untuk pengiriman titik akhir aliran data.

> **Note**
>
> Kuota default untuk jumlah grup keamanan yang dilampirkan per ENI adalah 5. Ini adalah batas yang dapat disesuaikan hingga 16, lihat Kuota VPC Amazon.

Peran IAM - Peran IAM adalah bagian dari Dataflow Endpoint dan tidak ditampilkan dalam diagram. Peran IAM yang digunakan untuk membuat dan melampirkan ENI lintas akun ke instans AWS Ground Station Amazon EC2.

Grup Keamanan 1 - Grup keamanan ini dilampirkan ke ENI yang akan dikaitkan dengan EC2 instans Amazon di akun Anda. Ini perlu memungkinkan lalu lintas UDP dari Grup Keamanan 2 pada port yang ditentukan dalam Anda dataflow-endpoint-group.

Elastic Network Interface (ENI) 1 - Anda harus mengaitkan Security Group 1 dengan ENI ini dan menempatkannya di subnet.

Subnet - Anda harus memastikan bahwa setidaknya ada satu alamat IP yang tersedia per aliran data untuk EC2 instans Amazon di akun Anda. Untuk detail lebih lanjut tentang ukuran subnet lihat, blok CIDR Subnet

Grup Keamanan 2 - Grup keamanan ini direferensikan di Dataflow Endpoint. Grup keamanan ini akan dilampirkan ke ENI yang AWS Ground Station akan digunakan untuk menempatkan data ke akun Anda.

Wilayah - Untuk informasi selengkapnya tentang wilayah yang didukung untuk koneksi lintas wilayah, lihatGunakan pengiriman data lintas wilayah.

CloudFormation Template berikut menunjukkan cara membuat infrastruktur yang dijelaskan di bagian ini.

```
DataflowEndpointSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Security Group for AWS Ground Station registration of Dataflow
 Endpoint Groups
    VpcId: YourVpcId

AWSGroundStationSecurityGroupEgress:
  Type: AWS::EC2::SecurityGroupEgress
  Properties:
    GroupId: !Ref: DataflowEndpointSecurityGroup
    IpProtocol: udp
    FromPort: 55555
    ToPort: 55555
    CidrIp: 10.0.0.0/8
    Description: "Allow AWS Ground Station to send UDP traffic on port 55555 to the
 10/8 range."
```

```
InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
    VpcId: YourVpcId
    SecurityGroupIngress:
      - IpProtocol: udp
        FromPort: 55555
        ToPort: 55555
        SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
        Description: "Allow AWS Ground Station Ingress from
 DataflowEndpointSecurityGroup"

ReceiverSubnet:
  Type: AWS::EC2::Subnet
  Properties:
    # Ensure your CidrBlock will always have at least one available IP address per
 dataflow endpoint.
    # See https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html for
 subent sizing guidelines.
    CidrBlock: "10.0.0.0/24"
    Tags:
      - Key: "Name"
        Value: "AWS Ground Station - Dataflow endpoint Example Subnet"
      - Key: "Description"
        Value: "Subnet for EC2 instance receiving AWS Ground Station data"
    VpcId: !Ref ReceiverVPC
```

Siapkan dan konfigurasikan Amazon EC2

Konfigurasi EC2 instans Amazon Anda dengan benar diperlukan agar pengiriman VITA-49 Signal/IP data or VITA-49 Extension data/IP secara sinkron dikirimkan melalui AWS Ground Station Agen atau titik akhir aliran data. Tergantung pada kebutuhan spesifik Anda, Anda dapat menggunakan prosesor Front End (FE) atau Software Defined Radio (SDR) secara langsung pada instance yang sama, atau Anda mungkin perlu menggunakan instance tambahan EC2. Pemilihan dan pemasangan FE atau SDR Anda berada di luar cakupan panduan pengguna ini. Untuk informasi selengkapnya tentang format data tertentu, lihatAWS Ground Station antarmuka bidang data.

Untuk informasi tentang persyaratan layanan kami, silakan lihat Ketentuan AWS Layanan.

Perangkat Lunak Umum yang Disediakan

AWS Ground Station menyediakan perangkat lunak umum untuk memudahkan penyiapan EC2 instans Amazon Anda.

AWS Ground Station Agen

AWS Ground Station Agen menerima data downlink Digital Intermediate Frequency (DiGIF) dan mengeluarkan data yang didekripsi yang memungkinkan hal-hal berikut:

- Kemampuan downlink DiGIF dari 40 MHz hingga 400 MHz bandwidth.
- Tingkat tinggi, pengiriman data DigiF jitter rendah ke IP publik (IP AWS Elastis) di jaringan. AWS
- Pengiriman data yang andal menggunakan Forward Error Correction (FEC).
- Mengamankan pengiriman data menggunakan AWS KMS kunci terkelola pelanggan untuk enkripsi.

Untuk informasi selengkapnya, lihat Panduan Pengguna AWS Ground Station Agen.

Aplikasi titik akhir Dataflow

Aplikasi jaringan yang digunakan oleh AWS Ground Station untuk mengirim dan menerima data antara lokasi AWS Ground Station antena, dan EC2 instans Amazon Anda. Ini dapat digunakan untuk uplink dan downlink data.

Radio yang Ditetapkan Perangkat Lunak (SDR)

Software defined radio (SDR) yang dapat digunakan untuk memodulasi/mendemodulasi sinyal yang digunakan untuk berkomunikasi dengan satelit Anda.

AWS Ground Station Gambar Mesin Amazon (AMIs)

Untuk mengurangi waktu pembuatan dan konfigurasi penginstalan ini, AWS Ground Station juga menawarkan yang telah dikonfigurasi sebelumnya AMIs. Aplikasi jaringan endpoint AMIs dengan aliran data dan radio yang ditentukan perangkat lunak (SDR) tersedia untuk akun Anda setelah orientasi Anda selesai. Mereka dapat ditemukan di EC2 konsol Amazon dengan mencari groundstation di Amazon Machine Images pribadi (AMIs). AWS Ground Station Agen AMIs with bersifat publik dan dapat ditemukan di EC2 konsol Amazon dengan mencari groundstation di Amazon Machine Images publik (AMIs).

Bekerja dengan telemetri

AWS Ground Station telemetri memberikan metrik hampir real-time dari AWS Ground Station antena selama kontak satelit Anda. Anda dapat menggunakan data telemetri untuk memantau kinerja kontak, mendeteksi anomali, dan membuat keputusan berdasarkan informasi tentang komunikasi satelit Anda.

Topik

Bagaimana telemetri bekerja

Untuk menggunakan telemetri, Anda mengonfigurasi a TelemetrySinkConfigyang menentukan di mana AWS Ground Station harus mengirimkan data telemetri. Anda kemudian menambahkan konfigurasi ini ke profil misi Anda menggunakan `telemetrySinkConfigArn` bidang. Selama kontak yang menggunakan profil misi berkemampuan telemetri,AWS Ground Station mengalirkan data telemetri ke akun Anda.

Proses pengiriman telemetri bekerja sebagai berikut:

1. Anda membuat aliran Kinesis Data Streams AWS di akun Anda untuk menerima data telemetri. Aliran harus dibuat di akun dan wilayah yang sama tempat Anda menjadwalkan kontak Anda.
2. Anda membuat peran IAM yang memberikan AWS Ground Station izin untuk menulis data ke aliran Anda.
3. Anda membuat referensi TelemetrySinkConfig yang mereferensikan peran streaming dan IAM Anda.
4. Anda menambahkan TelemetrySinkConfig ke profil misi Anda.
5. Anda mencantumkan dan memesan kontak menggunakan profil misi berkemampuan telemetri baru.
6. Selama kontak menggunakan profil misi ini,AWS Ground Station mengalirkan data telemetri ke aliran Data Streams Kinesis Anda dalam waktu dekat.

7. Anda menggunakan dan memproses data telemetri dari aliran Anda menggunakan AWS layanan atau aplikasi Anda sendiri.

Jenis telemetri yang tersedia

AWS Ground Station menyediakan jenis telemetri berikut selama kontak:

> **Note**
>
> AWS Ground Station sedang berupaya memperluas jumlah jenis telemetri yang didukung

Menunjuk telemetri

Memberikan informasi tentang arah penunjuk antena selama kontak satelit. Jenis telemetri ini selalu dikirim selama kontak dan mencakup sudut azimuth dan elevasi aktual dan perintah. Untuk informasi selengkapnya, lihat Menunjuk telemetri.

Melacak telemetri

Memberikan informasi tentang status pelacakan antena dan kesalahan pelacakan. Jenis telemetri ini dikirim saat pelacakan otomatis diaktifkan di konfigurasi pelacakan Anda. Untuk informasi selengkapnya, lihat Melacak telemetri.

Ketersediaan wilayah

Telemetri tersedia di semua AWS Wilayah tempat AWS Ground Station beroperasi. Selama eksekusi kontak, telemetri akan dikirimkan dari AWS Ground Station antena ke wilayah tempat Anda menjadwalkan kontak Anda, memberikan dukungan lintas wilayah.

Untuk daftar lengkap AWS Ground Station Wilayah dan lokasi stasiun bumi, lihatAWS Ground Station Lokasi.

Siapkan telemetri

Ikuti langkah-langkah ini untuk mengonfigurasi telemetri untuk kontak Anda AWS Ground Station. Setelah menyelesaikan penyiapan ini, data telemetri akan dikirimkan ke aliran Kinesis Data Streams Anda selama kontak yang menggunakan profil misi berkemampuan telemetri. Untuk pemahaman

mendalam tentang Kinesis Data Streams, silakan merujuk ke Panduan Pengguna Kinesis Data Streams.

Langkah 1: Buat prasyarat AWS sumber daya

CloudFormation Cuplikan berikut menunjukkan cara membuat sumber daya prasyarat AWS untuk pengiriman telemetri. Cuplikan ini membuat aliran Kinesis Data Streams dan peran IAM yang memberikan izin untuk menulis data AWS Ground Station telemetri ke aliran.

```
TelemetryStream:
  Type: AWS::Kinesis::Stream
  Properties:
    Name: GroundStationTelemetryStream
    StreamModeDetails:
      StreamMode: ON_DEMAND
    RetentionPeriodHours: 24

TelemetryRole:
  Type: AWS::IAM::Role
  Properties:
    RoleName: GroundStationTelemetryRole
    AssumeRolePolicyDocument:
      Version: '2012-10-17'
      Statement:
        - Effect: Allow
          Principal:
            Service: groundstation.amazonaws.com
          Action: sts:AssumeRole
    Policies:
      - PolicyName: KinesisWritePolicy
        PolicyDocument:
          Version: '2012-10-17'
          Statement:
            - Effect: Allow
              Action:
                - kinesis:DescribeStream
                - kinesis:PutRecord
                - kinesis:PutRecords
              Resource: !GetAtt TelemetryStream.Arn
```

Daftar di bawah ini menyebutkan pertimbangan pengaturan unik saat mengonfigurasi pengiriman telemetri untuk.AWS Ground Station

Aliran Data Streams Kinesis - Aliran menggunakan mode kapasitas sesuai permintaan, yang secara otomatis menskalakan berdasarkan throughput. Ini direkomendasikan untuk sebagian besar kasus penggunaan. Aliran dikonfigurasi untuk menyimpan data selama 24 jam. Secara default, aliran menggunakan enkripsi AWS terkelola. Untuk menggunakan enkripsi yang dikelola pelanggan AWS Key Management Service, tambahkan `StreamEncryption` properti dan perbarui kebijakan peran IAM untuk menyertakan izin. `kms:GenerateDataKey` Untuk informasi selengkapnya, lihat Perlindungan Data di Amazon Kinesis Data Streams.

Peran IAM - Peran IAM memungkinkan kepala `groundstation.amazonaws.com` layanan untuk mengambil peran dan menulis data telemetri ke aliran Kinesis Data Streams Anda. Kebijakan peran memberikan izin untuk`kinesis:DescribeStream`,`kinesis:PutRecord`, dan `kinesis:PutRecords` tindakan di streaming. Lihat Konfigurasi Wastafel Telemetri panduan tentang pengaturan kebijakan kepercayaan dan kebijakan peran.

Konfigurasi tambahan - Tambahkan `iam:PassRole` izin ke pengguna IAM atau peran yang Anda gunakan untuk panggilan AWS Ground Station API. Ini memungkinkan Anda untuk meneruskan peran telemetri AWS Ground Station saat membuat file. TelemetrySinkConfig

Contoh PassRole Kebijakan

Untuk informasi selengkapnya tentang cara memperbarui atau melampirkan kebijakan peran, lihat Mengelola kebijakan IAM di Panduan Pengguna IAM. Untuk informasi selengkapnya tentang `iam:PassRole` izin, lihat Memberikan izin pengguna untuk meneruskan peran ke layanan AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::999999999999:role/your-telemetry-delivery-role-name"
    }
  ]
}
```

Langkah 2: Buat TelemetrySinkConfig

Buat TelemetrySinkConfigyang menentukan bagaimana AWS Ground Station akan mengirimkan data telemetri ke aliran Kinesis Data Streams Anda. Gunakan arus ARN dan peran ARN dari output CloudFormation tumpukan di Langkah 1.

> **Note**
>
> Saat Anda membuat TelemetrySinkConfig,AWS Ground Station akan memverifikasi akses ke aliran Kinesis Data Streams Anda dengan mengirimkan catatan pengujian kosong dengan kunci partisi. `test`

Untuk informasi selengkapnya tentang membuat TelemetrySinkConfig, lihatKonfigurasi Wastafel Telemetri.

Langkah 3: Tambahkan telemetri ke profil misi Anda

Buat profil misi. Untuk informasi selengkapnya tentang membuat profil misi, lihatGunakan Profil AWS Ground Station Misi. Tambahkan `telemetrySinkConfigArn` ke profil misi Anda untuk mengaktifkan pengiriman telemetri selama kontak. Gunakan ARN yang TelemetrySinkConfigdibuat di Langkah 2.

Langkah 4: Jadwalkan kontak

Jadwalkan kontak menggunakan profil misi berkemampuan telemetri Anda. Selama kontak,AWS Ground Station akan mengalirkan data telemetri ke aliran Kinesis Data Streams Anda.

Apa yang diharapkan selama kontak

- Telemetri mulai - Data mulai streaming saat kontak dimulai.
- Pengiriman mendekati waktu nyata - Telemetri tiba di aliran Kinesis Data Streams Anda dalam waktu dekat.
- Durasi kontak - Data berlanjut di seluruh kontak.
- Berhenti otomatis - Telemetri berhenti streaming saat kontak berakhir.

Memantau pengiriman

Anda dapat memantau pengiriman telemetri menggunakan:

- Metrik aliran Kinesis Data Streams - Periksa catatan masuk. CloudWatch Untuk informasi selengkapnya, lihat Memantau Amazon Kinesis Data Streams.
- Log aplikasi - Verifikasi pemrosesan data dalam aplikasi Anda yang mengkonsumsi dari aliran.
- Kinesis Data Viewer - Gunakan konsol aliran Kinesis Data Streams untuk melihat rekaman sampel dari aliran Anda.

Langkah selanjutnya

Setelah menyelesaikan pengaturan, Anda dapat:

- Pelajari tentang format data telemetri dan jenis telemetri yang tersedia. Lihat Memahami data telemetri.
- Buat aplikasi untuk memproses data telemetri dari aliran Kinesis Data Streams Anda. Untuk informasi selengkapnya, lihat Membangun Konsumen untuk Amazon Kinesis Data Streams.
- Buat dasbor dan peringatan menggunakan CloudWatch dan layanan lainnya AWS.
- Tinjau panduan pemecahan masalah jika Anda mengalami masalah. Lihat Memecahkan masalah telemetri.

Memahami data telemetri

Data telemetri dikirimkan sebagai catatan Base64-encoded JSON ke aliran Kinesis Data Streams Anda. Setiap catatan berisi informasi yang dikumpulkan selama kontak satelit Anda, termasuk metadata tentang kontak dan pengukuran telemetri sampel.

Ikhtisar format data

Setiap catatan telemetri berisi komponen-komponen berikut:

Jenis dan versi telemetri

Mengidentifikasi jenis data telemetri tertentu dan versi skemanya. Ini memungkinkan Anda untuk mengurai berbagai jenis telemetri dengan tepat. Untuk informasi selengkapnya tentang pembuatan versi skema, lihat. Pembuatan versi dan evolusi skema

ID Lingkup

Pengidentifikasi unik untuk ruang lingkup telemetri. Ini memungkinkan Anda untuk menghubungkan data telemetri dengan kontak tertentu.

Metadata

Informasi kontekstual tentang telemetri.

Data

Pengukuran telemetri sampel khusus untuk jenis telemetri.

Kunci partisi

Rekaman telemetri dikirim ke aliran Kinesis Data Streams Anda dengan kunci partisi dalam format:

```
SCOPE#scopeId#TELEMETRY_ID#telemetryId#TELEMETRY_VERSION#telemetryVersion
```

Kunci partisi ini memastikan bahwa semua telemetri dari jenis tertentu untuk satu kontak dikirim ke pecahan yang sama dalam aliran Kinesis Data Streams Anda, memberikan upaya terbaik untuk pemesanan aliran telemetri kontak tersebut.

Menunjuk telemetri

Telemetri penunjuk memberikan informasi tentang arah penunjuk antena selama kontak satelit. Jenis telemetri ini selalu dikirim selama kontak.

Bidang data

SampleTimeStamp

Waktu ketika data telemetri diambil sampelnya, dalam ISO-8601 format dalam UTC dengan presisi milidetik.

azimuth

Sudut azimuth aktual antena dalam derajat.

elevasi

Sudut elevasi aktual antena dalam derajat.

CommandeDazimuth

Sudut azimuth yang diperintahkan dalam derajat. Ini adalah sudut azimuth target yang coba dicapai antena.

CommandeDelevation

Sudut elevasi yang diperintahkan dalam derajat. Ini adalah sudut elevasi target yang coba dicapai antena.

> **Note**
>
> Posisi antena sebenarnya mungkin berbeda dari posisi yang diperintahkan karena keterbatasan fisik atau penundaan mekanis selama kontak.

Bidang metadata

GroundStation

Nama stasiun bumi (misalnya, “Ohio 1").

SatelliteID

Pengidentifikasi sumber daya satelit di AWS Ground Station.

contactId

Pengidentifikasi kontak.

Contoh JSON

```
{
  "telemetryTypeAndVersion": "POINTING#1.0.0",
  "telemetryType": "POINTING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
    "satelliteId": "87654321-4321-4321-4321-210987654321",
    "contactId": "12345678-1234-1234-1234-123456789012"
  },
  "data": {
    "sampleTimestamp": "2025-12-08T12:00:00.123Z",
    "azimuth": 180.5,
    "elevation": 45.2,
```

```
    "commandedAzimuth": 180.0,
    "commandedElevation": 45.0
  }
}
```

Melacak telemetri

Telemetri pelacakan memberikan informasi tentang status pelacakan antena dan kesalahan pelacakan. Jenis telemetri ini dikirim saat pelacakan otomatis diaktifkan di konfigurasi pelacakan Anda dan saat antena secara aktif menggunakan autotrack.

> **Note**
>
> Jika `autotrack` parameter dalam Anda TrackingConfig disetel ke`REMOVED`, tidak ada telemetri pelacakan yang akan dikirimkan. Untuk informasi selengkapnya tentang melacak konfigurasi, lihatMelacak Config.

Bidang data

SampleTimeStamp

Waktu ketika data telemetri diambil sampelnya, dalam ISO-8601 format dalam UTC dengan presisi milidetik.

Status Pelacakan

Status pelacakan antena saat ini. Kemungkinan nilainya adalah:

- `TRACKING`- Antena telah berhasil mengunci sinyal yang cocok dengan profil misi dan secara aktif mengikutinya melintasi langit. Ini adalah keadaan operasional nominal selama kontak.
- `ACQUIRING`- Antena sedang dalam proses mencari dan mengunci sinyal. Sistem saat ini menggunakan pelacakan terprogram, menunjuk berdasarkan data ephemeris.
- `MASKED`Posisi satelit yang diprediksi berada di belakang topeng autotrack, yang berarti antena tidak dapat menggunakan autotrack secara andal pada arah penunjuk tertentu. Ini biasanya terjadi pada area gangguan RF tinggi seperti elevasi rendah.

pelacakan ErrorAzimuth

Kesalahan pelacakan pada sumbu azimuth, diukur dalam derajat.

pelacakan ErrorElevation

Kesalahan pelacakan pada sumbu elevasi, diukur dalam derajat.

> **Note**
>
> Nilai kesalahan pelacakan mewakili penyesuaian dari trek program berbasis ephemeris yang AWS Ground Station berlaku selama pelacakan otomatis untuk memaksimalkan kekuatan sinyal.

Bidang metadata

Telemetri pelacakan mencakup bidang metadata yang sama dengan telemetri penunjuk:,, dan. `groundStation` `satelliteId` `contactId`

Contoh JSON

```
{
  "telemetryTypeAndVersion": "TRACKING#1.0.0",
  "telemetryType": "TRACKING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
    "satelliteId": "87654321-4321-4321-4321-210987654321",
    "contactId": "12345678-1234-1234-1234-123456789012"
  },
  "data": {
    "sampleTimestamp": "2025-12-08T12:00:00.123Z",
    "trackingStatus": "TRACKING",
    "trackingErrorAzimuth": 0.2,
    "trackingErrorElevation": 0.1
  }
}
```

Membaca data dari aliran Kinesis Data Streams

Data telemetri dikirimkan ke aliran Kinesis Data Streams Anda dan dapat dikonsumsi menggunakan pola konsumsi aliran standar. Saat membaca data dari streaming Anda, ingatlah pertimbangan berikut.

Penguraian kode Base64

Data dalam aliran Kinesis Data Base64-encoded Streams adalah. Anda harus memecahkan kode data sebelum menguraikannya sebagai JSON. Untuk informasi selengkapnya, lihat Bekerja dengan Amazon Kinesis Data Streams.

Menggunakan Penampil Data Kinesis

Untuk akses cepat ke data telemetri Anda, konsol aliran Kinesis Data Streams menawarkan fitur Penampil Data. Saat menggunakan fitur ini:

- Pengiriman telemetri dapat terjadi pada pecahan apa pun dalam aliran Anda.
- Posisi awal default dibaca dari catatan terbaru di pecahan.
- Anda mungkin perlu menyesuaikan pecahan yang dipilih dan menggunakan posisi awal “Pada stempel waktu” untuk melihat catatan yang diterima.

Menggunakan Perpustakaan Klien Kinesis

Perpustakaan Klien Kinesis (KCL) mengelola banyak kompleksitas yang terkait dengan konsumsi data dari aliran Kinesis Data Streams, termasuk manajemen shard, checkpointing, dan load balancing. Kami merekomendasikan penggunaan KCL untuk aplikasi konsumsi telemetri produksi.

Untuk informasi selengkapnya, lihat Mengembangkan Konsumen Menggunakan Perpustakaan Klien Kinesis.

Praktik terbaik untuk konsumsi

- Minimalkan latensi - Gunakan Enhanced Fan-Out untuk membaca dari aliran Kinesis Data Streams dengan throughput khusus dan latensi yang lebih rendah dibandingkan dengan polling. Untuk informasi selengkapnya, lihat Mengembangkan Fan-Out Konsumen yang Ditingkatkan.
- Aliran khusus - Gunakan aliran Kinesis Data Streams khusus AWS Ground Station untuk integrasi telemetri Anda. Berbagi aliran dengan aplikasi lain dapat menyebabkan saturasi throughput tulis dan kegagalan pengiriman telemetri.
- On-demand kapasitas - Terapkan aliran Kinesis Data Streams Anda dalam mode penyediaan sesuai permintaan untuk memungkinkan penskalaan pecahan otomatis berdasarkan throughput.
- Monitor throughput - Pantau streaming Anda untuk pembatasan menggunakan metrik. CloudWatch Untuk informasi selengkapnya, lihat Memantau Amazon Kinesis Data Streams.

Pembuatan versi dan evolusi skema

Skema telemetri diberi versi untuk mendukung evolusi dari waktu ke waktu. `telemetryVersion`Bidang di setiap catatan menunjukkan versi skema.

Menangani perubahan skema

- Jenis telemetri baru dapat diperkenalkan di masa depan.
- Jenis telemetri yang ada mungkin menerima versi baru dengan perubahan yang melanggar.
- Aplikasi Anda harus toleran terhadap jenis dan versi telemetri yang tidak diketahui.
- Menguraitelemetry`TypeAndVersion`,`telemetryType`, dan `telemetryVersion` bidang untuk menentukan bagaimana memproses setiap catatan.

Sebaiknya terapkan serialisasi payload sadar versi yang dapat menangani beberapa versi skema dengan anggun, memungkinkan aplikasi Anda terus berfungsi saat versi baru diperkenalkan.

Bekerja dengan kontak

Anda dapat memasukkan data satelit, mengidentifikasi lokasi antena, berkomunikasi, dan menjadwalkan waktu antena untuk satelit yang dipilih dengan menggunakan AWS Ground Station konsol AWS CLI, atau AWS SDK dalam bahasa pilihan Anda. Anda dapat meninjau, membatalkan, dan menjadwal ulang reservasi kontak hingga 15 menit sebelum kontak mulai*. Anda juga dapat memperbarui kontak untuk menentukan penggantian ephemeris — termasuk azimuth/elevation, data pelacakan OEM, atau TLE — atau mengubah satelit target. Untuk informasi selengkapnya, lihat Perbarui kontak dan versi kontak. Selain itu, Anda dapat melihat detail paket harga menit cadangan Anda jika Anda menggunakan model harga menit yang AWS Ground Station dipesan.

AWS Ground Station mendukung pengiriman data lintas wilayah. Konfigurasi titik akhir aliran data yang merupakan bagian dari profil misi yang Anda pilih menentukan wilayah mana data dikirimkan. Untuk informasi selengkapnya tentang penggunaan pengiriman data lintas wilayah, lihatGunakan pengiriman data lintas wilayah.

Untuk menjadwalkan kontak, sumber daya Anda harus dikonfigurasi. Jika Anda belum mengonfigurasi sumber daya Anda, lihatMemulai. Ketika ReserveContactdipanggil,AWS Ground Station mengambil snapshot dari profil misi dan sumber daya konfigurasi untuk digunakan di seluruh siklus hidup kontak. Perubahan pada sumber daya ini menggunakan UpdateConfigAPI UpdateMissionProfiledan tidak akan tercermin dalam kontak yang dicadangkan sebelum pembaruan. Jika Anda memerlukan perubahan sumber daya yang diterapkan ke kontak yang sudah dijadwalkan, Anda harus terlebih dahulu membatalkan kontak menggunakan CancelContact, dan kemudian menjadwal ulang menggunakan ReserveContact.

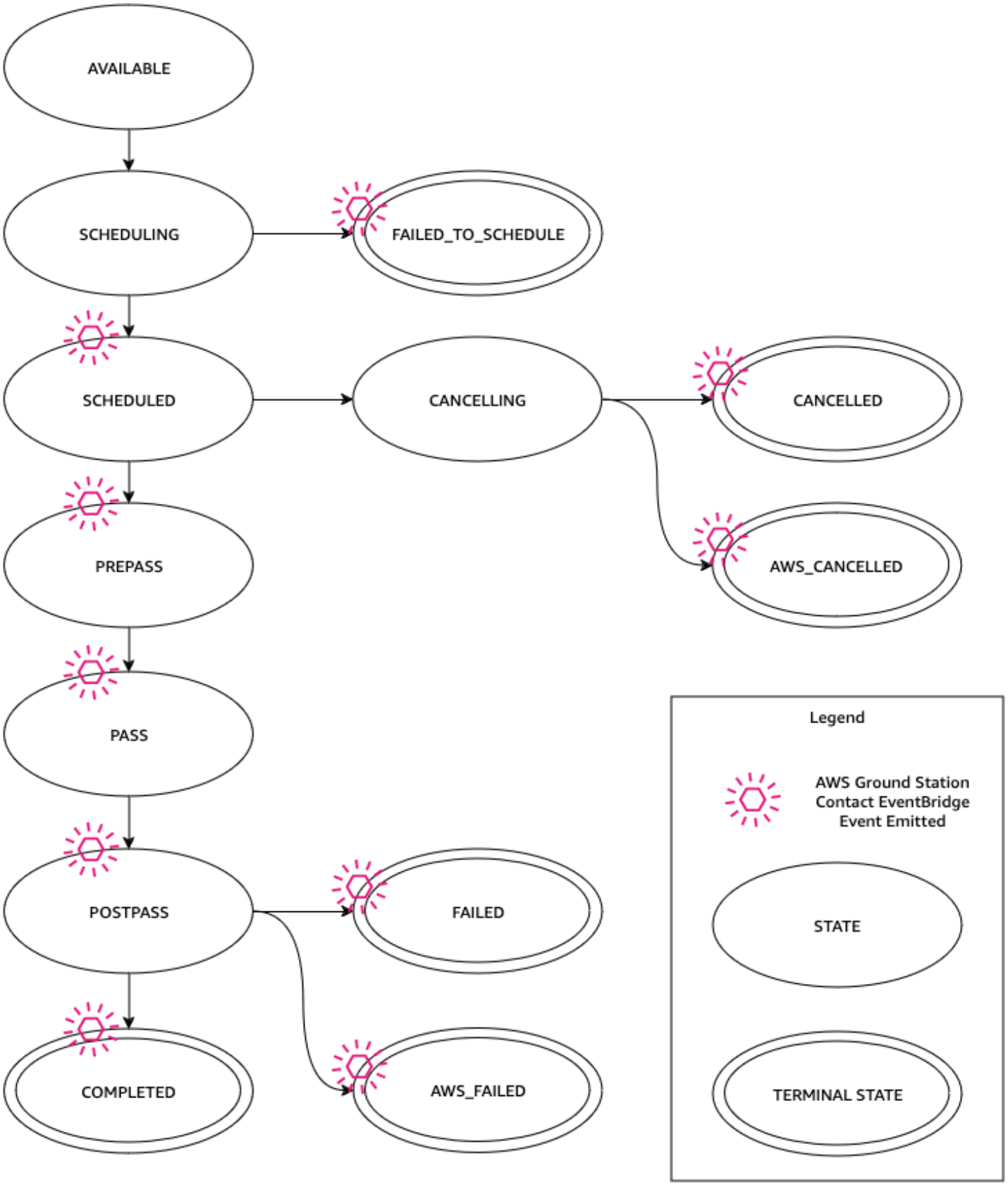
* Kontak yang dibatalkan dapat dikenakan biaya ketika dibatalkan terlalu dekat dengan waktu kontak. Untuk informasi selengkapnya tentang kontak yang dibatalkan, lihat: FAQ Ground Station.

Topik

Memahami siklus hidup kontak

Memahami siklus hidup kontak dapat membantu Anda mengotomatisasi dan memecahkan masalah berbagai masalah saat menggunakan.AWS Ground Station Diagram berikut menunjukkan siklus

hidup AWS Ground Station kontak serta Event Bridge Event yang dipancarkan selama siklus hidup. Penting untuk dicatat bahwa COMPLETED, FAILED, FAILED_TO_SCHEDULE, CANCELLED, AWS_CANCELLED, dan AWS_FAILED adalah status terminal. Kontak tidak akan bertransisi keluar dari status terminal. Lihat detail tentang apa yang ditunjukkan setiap status dan apakah status tersebut dapat dihentikan atau dibatalkan menggunakan. AWS Ground Station status kontak CancelContact


AVAILABLE
SCHEDULING
FAILED_TO_SCHEDULE
SCHEDULED
CANCELLING
CANCELLED
PREPASS
AWS_CANCELLED
PASS
POSTPASS
FAILED
COMPLETED
AWS_FAILED
Legend
AWS Ground Station Contact EventBridge Event Emitted
STATE
TERMINAL STATE

AWS Ground Station status kontak

Status AWS Ground Station kontak memberikan wawasan tentang apa yang terjadi pada kontak itu pada waktu tertentu.

Status kontak

Tabel berikut menjelaskan status yang dapat dimiliki kontak:

Status	Deskripsi	Terminal	Dibatalkan	Dapat dihentikan
AVAILABLE	Kontak tersedia untuk dipesan.	Tidak	N/A	N/A
PENJADWALAN	Kontak sedang dalam proses penjadwalan.	Tidak	Ya	Tidak
DIJADWALKAN	Kontak berhasil dijadwalkan.	Tidak	Ya	Tidak
FAILED_TO _SCHEDULE	Kontak gagal menjadwalkan.	Ya	Tidak	Tidak
PREPASS	Kontak akan segera dimulai dan sumber daya sedang dipersiap kan.	Tidak	Ya	Tidak
LULUS	Kontak saat ini sedang dijalanka n dan satelit sedang dikomunik asikan.	Tidak	Tidak	Ya
POSTPASS	Komunikasi telah selesai dan sumber daya yang digunakan sedang dibersihkan.	Tidak	Tidak	Tidak
DISELESAIKAN	Kontak selesai tanpa kesalahan.	Ya	Tidak	Tidak
FAILED	Kontak gagal karena masalah dengan konfigurasi sumber daya Anda.	Ya	Tidak	Tidak

Status	Deskripsi	Terminal	Dibatalkan	Dapat dihentikan
AWS_FAILED	Kontak gagal karena masalah dalam AWS Ground Station layanan.	Ya	Tidak	Tidak
MEMBATALKAN	Kontak sedang dalam proses dibatalkan.	Tidak	Tidak	Tidak
AWS_DIBATALKAN	Kontak dibatalkan oleh AWS Ground Station layanan. Antena atau pemeliharaan situs, dan penyimpangan ephemeris adalah contoh kapan ini bisa terjadi.	Ya	Tidak	Tidak
DIBATALKAN	Kontak dibatalkan oleh Anda.	Ya	Tidak	Tidak

> **Note**
>
> Untuk informasi tentang implikasi penagihan dari kontak yang dibatalkan atau dihentikan, lihat. Memahami penagihan kontak

Retensi Data Kontak

AWS Ground Station menyimpan data kontak selama 1 tahun setelah ReserveContactpermintaan dibuat untuk memesan kontak. Setelah periode 1 tahun, data kontak dihapus.

Jika Anda perlu menyimpan data kontak lebih dari satu tahun, disarankan untuk mengekspor data Anda sebelum periode penyimpanan berakhir. Untuk informasi lebih lanjut tentang cara mengakses dan mengekspor data kontak, lihat:

- AWS Ground Station Referensi API
- AWS Ground Station Referensi Perintah CLI

Memahami penagihan kontak

Dengan AWS Ground Station, Anda hanya membayar untuk waktu antena yang Anda gunakan.AWS Ground Station penggunaan kontak meter per menit. Untuk setiap kontak, layanan menghitung durasi kontak dari waktu mulai hingga akhir dan membulatkan hingga menit terdekat. Durasi terukur ini menentukan biaya Anda untuk kontak itu.

Tarif Anda tergantung pada dua faktor utama:

- Bandwidth — Jumlah bandwidth yang disediakan untuk kontak (narrowband atau wideband)
- Lokasi stasiun darat - Tarif bervariasi menurut lokasi stasiun bumi

Definisi bandwidth

AWS Ground Station mengkategorikan kontak menjadi dua tingkatan bandwidth berdasarkan bandwidth sesaat:

- Narrowband — Setiap kontak di mana bandwidth sesaat kurang dari atau sama dengan 40 MHz
- Wideband - Setiap kontak di mana bandwidth sesaat lebih besar dari 40 MHz

Mode penjadwalan

AWS Ground Station menawarkan dua mode penjadwalan:

- On-Demand— Bayar akses antena tanpa komitmen jangka panjang
- Cadangan — Memberikan tarif diskon dan penjadwalan yang lebih baik dibandingkan dengan On-Demand, dengan komitmen bulanan. Harga menit cadangan tersedia untuk pelanggan yang berkomitmen untuk penggunaan bulanan untuk jangka waktu tertentu.

Untuk informasi harga spesifik untuk akun Anda atau untuk mempelajari lebih lanjut tentang mode penjadwalan Cadangan, hubungi perwakilan AWS Anda.

CancelContact

Penggunaan CancelContactAPI bervariasi berdasarkan status kontak saat Anda memanggilnya:

- Sebelum kontak mulai - Membatalkan kontak sepenuhnya

- Setelah kontak dimulai dan sebelum kontak berakhir - Menghentikan kontak yang sedang berlangsung

Ketika Anda membatalkan kontak, penagihan tergantung pada mode penjadwalan Anda dan kapan Anda membatalkan. Untuk informasi selengkapnya, hubungi perwakilan AWS Anda.

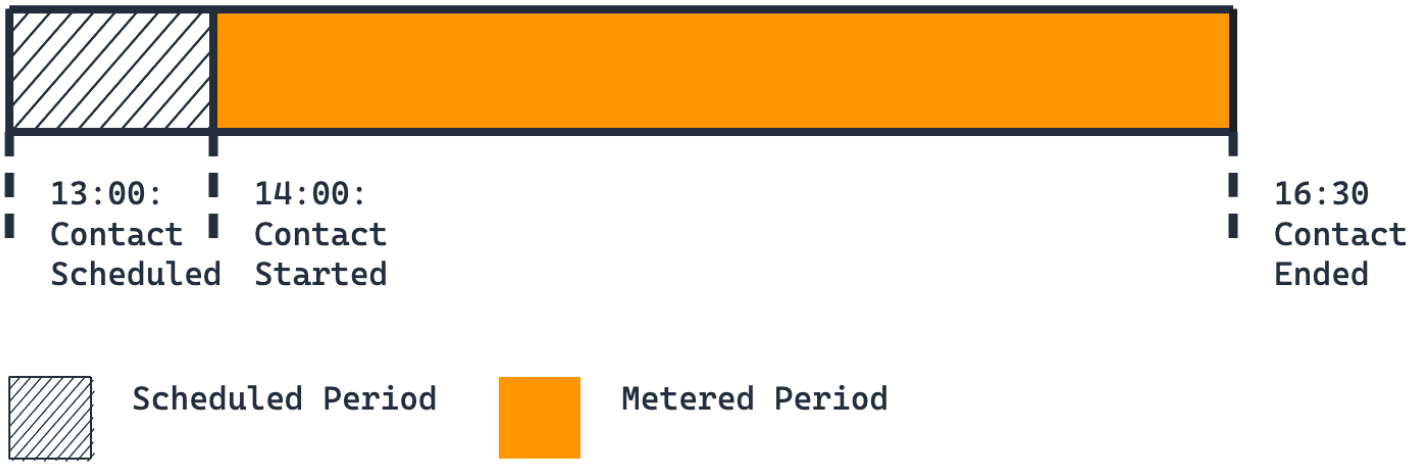
Ketika Anda menghentikan kontak, Anda ditagih untuk bagian kontak yang dieksekusi, dan sisa waktu yang tidak tercakup oleh kontak duplikat. Kontak duplikat dalam konteks ini adalah:

- Dijadwalkan di stasiun bumi yang sama dengan kontak berhenti asli
- Dijadwalkan dengan ID akun AWS yang sama dengan kontak asli yang dihentikan
- Dicadangkan setelah perintah dikeluarkan untuk menghentikan kontak asli

Skenario berikut menunjukkan bagaimana pengukuran ini bekerja dalam praktik.

Skenario 1: Kontak tunggal

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30.



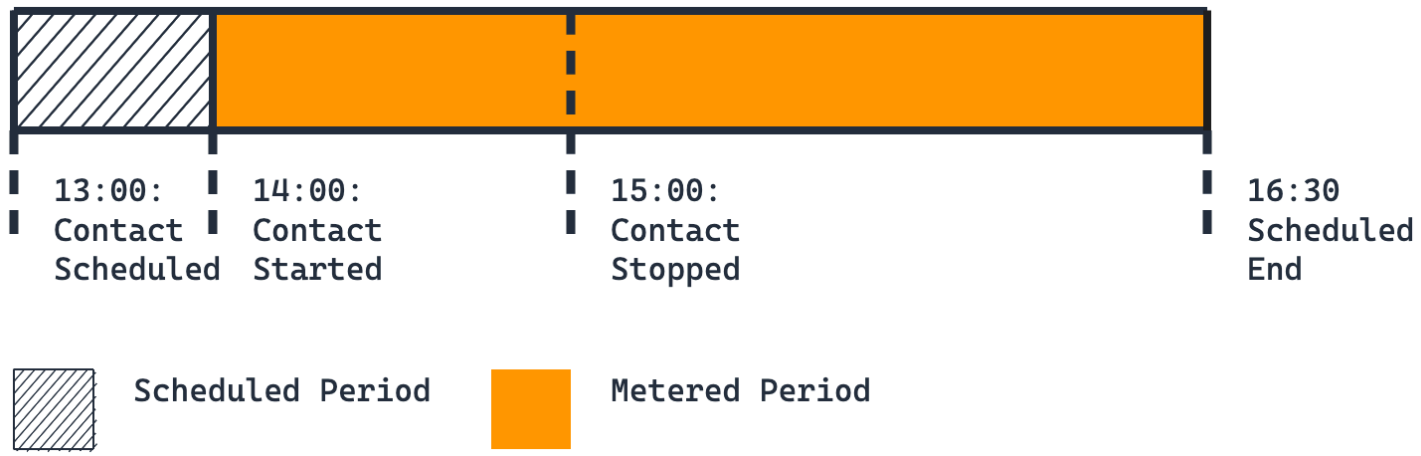
Rincian penagihan:

- Kontak pertama: 150 menit (durasi penuh)

Anda ditagih selama 150 menit. Ini adalah skenario dasar di mana kontak berjalan hingga penyelesaiannya yang dijadwalkan tanpa berhenti atau pembatalan.

Skenario 2: Kontak berhenti tunggal

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pukul 15:00, Anda menelepon CancelContact API untuk menghentikan kontak Anda.



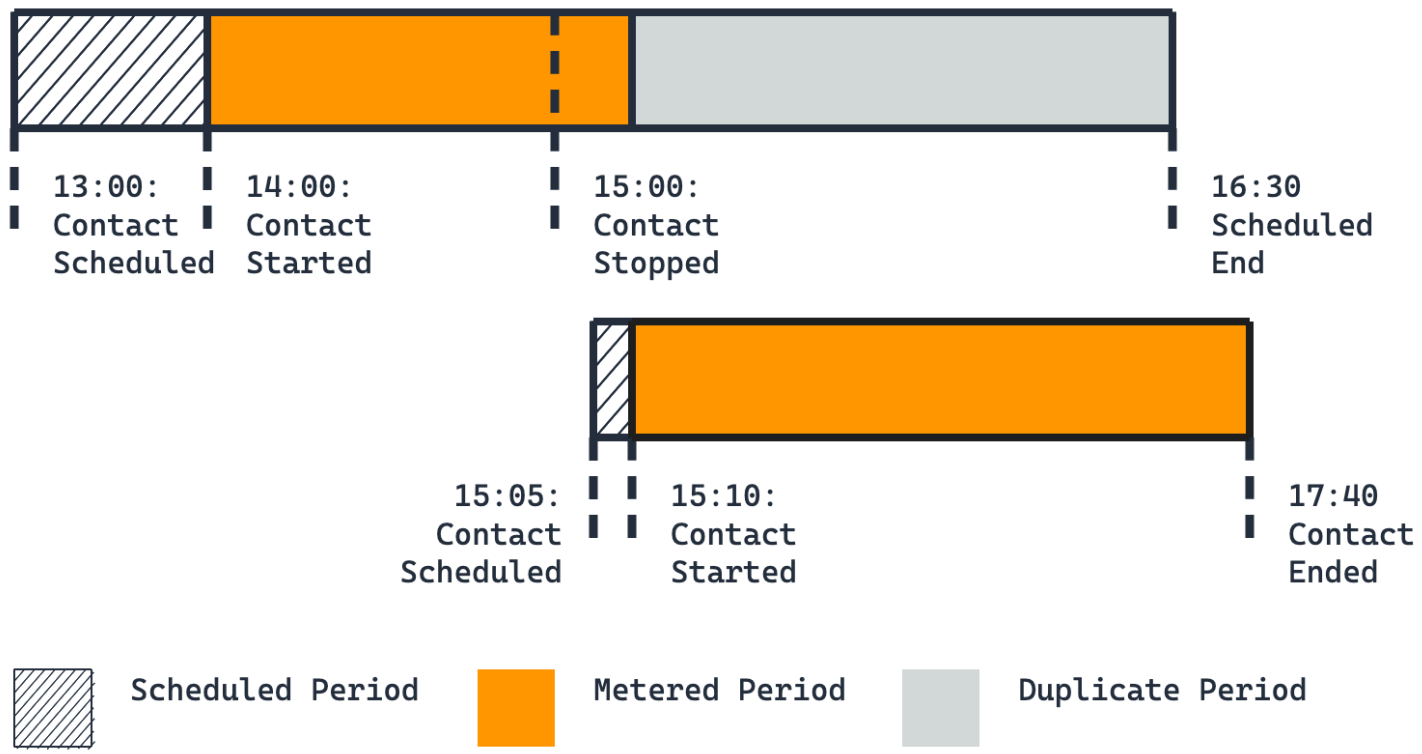
Rincian penagihan:

- Kontak pertama: 150 menit (durasi asli penuh)

Anda ditagih selama 150 menit penuh karena Anda menghentikan kontak tetapi tidak menjadwalkan kontak duplikat untuk menutupi sisa waktu (15:00-16:30). Ketika Anda menghentikan kontak tanpa menjadwalkan duplikat, Anda tetap bertanggung jawab atas seluruh durasi yang dijadwalkan semula.

Skenario 3: Duplikat tunggal

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Setelah menelepon CancelContact, Anda menjadwalkan kontak lain di Ground Station yang sama mulai pukul 15:10 selama 150 menit.



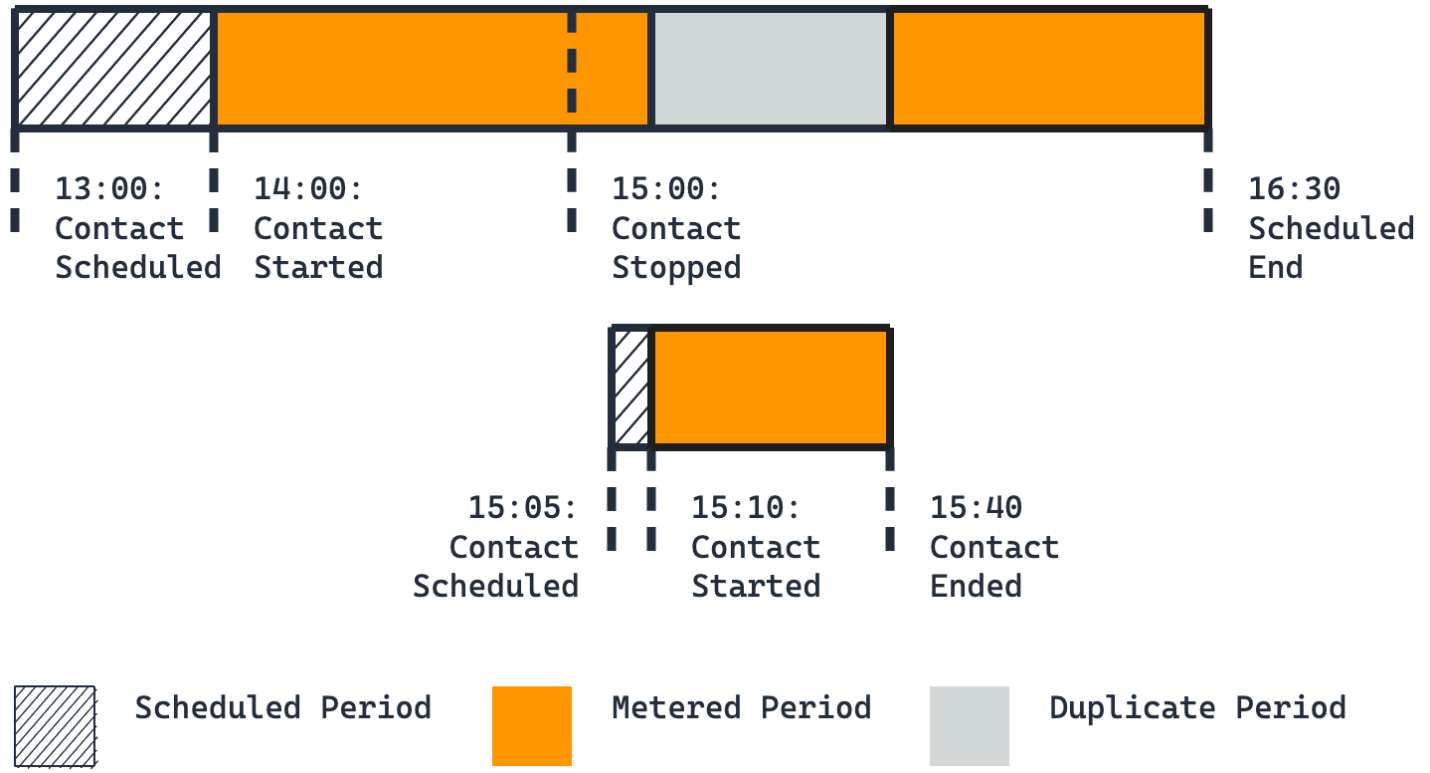
Rincian penagihan:

- Kontak pertama: 70 menit (60 menit dieksekusi +10 menit downtime sebelum kontak kedua dimulai)
- Kontak kedua: 150 menit (durasi penuh)

Kontak kedua adalah duplikat karena Anda menjadwalkannya setelah menghentikan kontak pertama. Duplikat mencakup sisa waktu dari 15:10 hingga 16:30, jadi Anda hanya ditagih untuk saat kontak pertama benar-benar berjalan ditambah jarak 10 menit antara berhenti dan memulai kembali.

Skenario 4: Duplikat pendek

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Setelah menelepon CancelContact, Anda menjadwalkan kontak 30 menit di Ground Station yang sama mulai pukul 15:10.



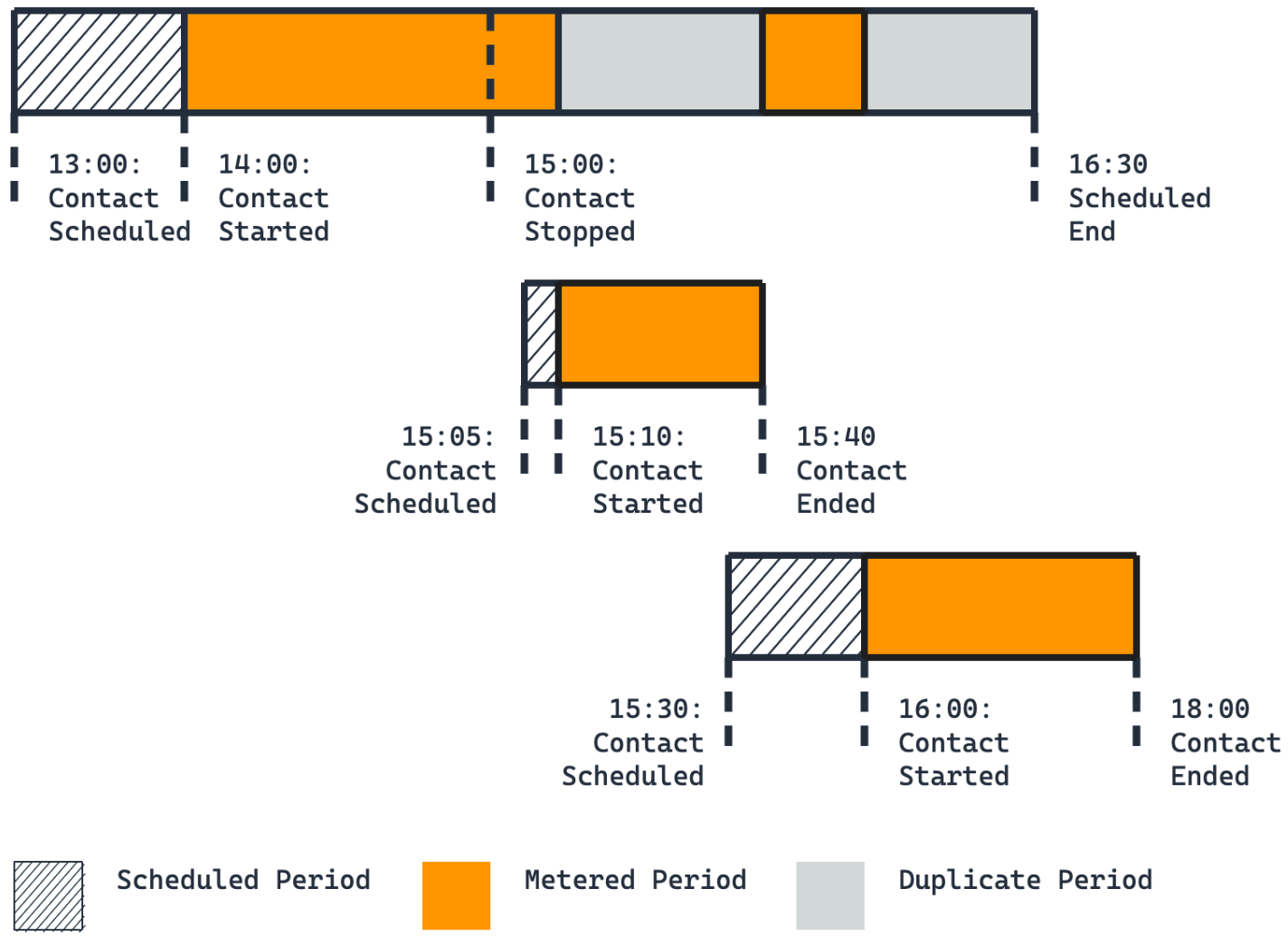
Rincian penagihan:

- Kontak pertama: 120 menit (60 menit dieksekusi +10 menit downtime sebelum kontak kedua dimulai+50 menit dari sisa waktu yang tidak tercakup oleh duplikat)
- Kontak kedua: 30 menit (durasi penuh)

Kontak duplikat hanya mencakup 30 menit (15:10-15:40) dari 90 menit tersisa setelah Anda menghentikan kontak pertama. Anda ditagih untuk kedua celah 10 menit sebelum duplikat dimulai dan 50 menit waktu terbuka setelah duplikat berakhir (15:40-16:30).

Skenario 5: Beberapa duplikat

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Setelah menelepon CancelContact, Anda menjadwalkan kontak 30 menit di Ground Station yang sama mulai pukul 15:10. Kemudian, pukul 15:30, Anda menjadwalkan kontak lain mulai pukul 16:00 selama 120 menit.



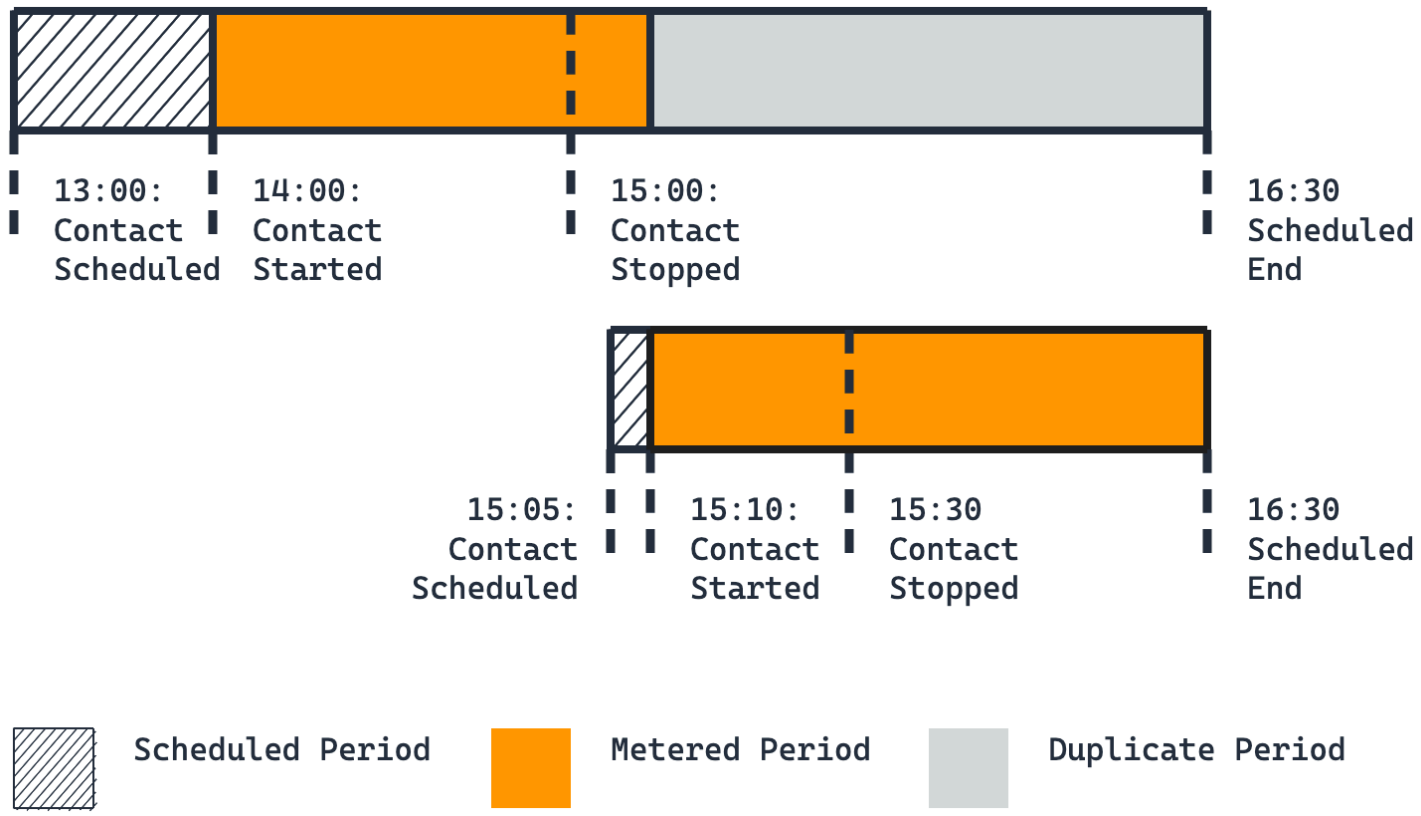
Rincian penagihan:

- Kontak pertama: 90 menit (60 menit dieksekusi +10 menit downtime sebelum kontak kedua dimulai +20 menit downtime antara kontak kedua dan ketiga)
- Kontak kedua: 30 menit (durasi penuh)
- Kontak ketiga: 120 menit (durasi penuh)

Kedua kontak kedua dan ketiga dihitung sebagai duplikat karena Anda menjadwalkannya setelah menghentikan kontak pertama. Namun, Anda masih ditagih untuk kesenjangan antara kontak: 10 menit antara pemberhentian pertama (15:00) dan awal kedua (15:10), dan 20 menit antara akhir kedua (15:40) dan awal ketiga (16:00).

Skenario 6: Beberapa berhenti

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Setelah menelepon CancelContact, Anda menjadwalkan kontak 80 menit di Ground Station Anytown 1 yang dimulai pukul 15:10 dan berakhir pukul 16:30. Pada 15:30, Anda memanggil CancelContact API lagi, menghentikan kontak duplikat Anda.



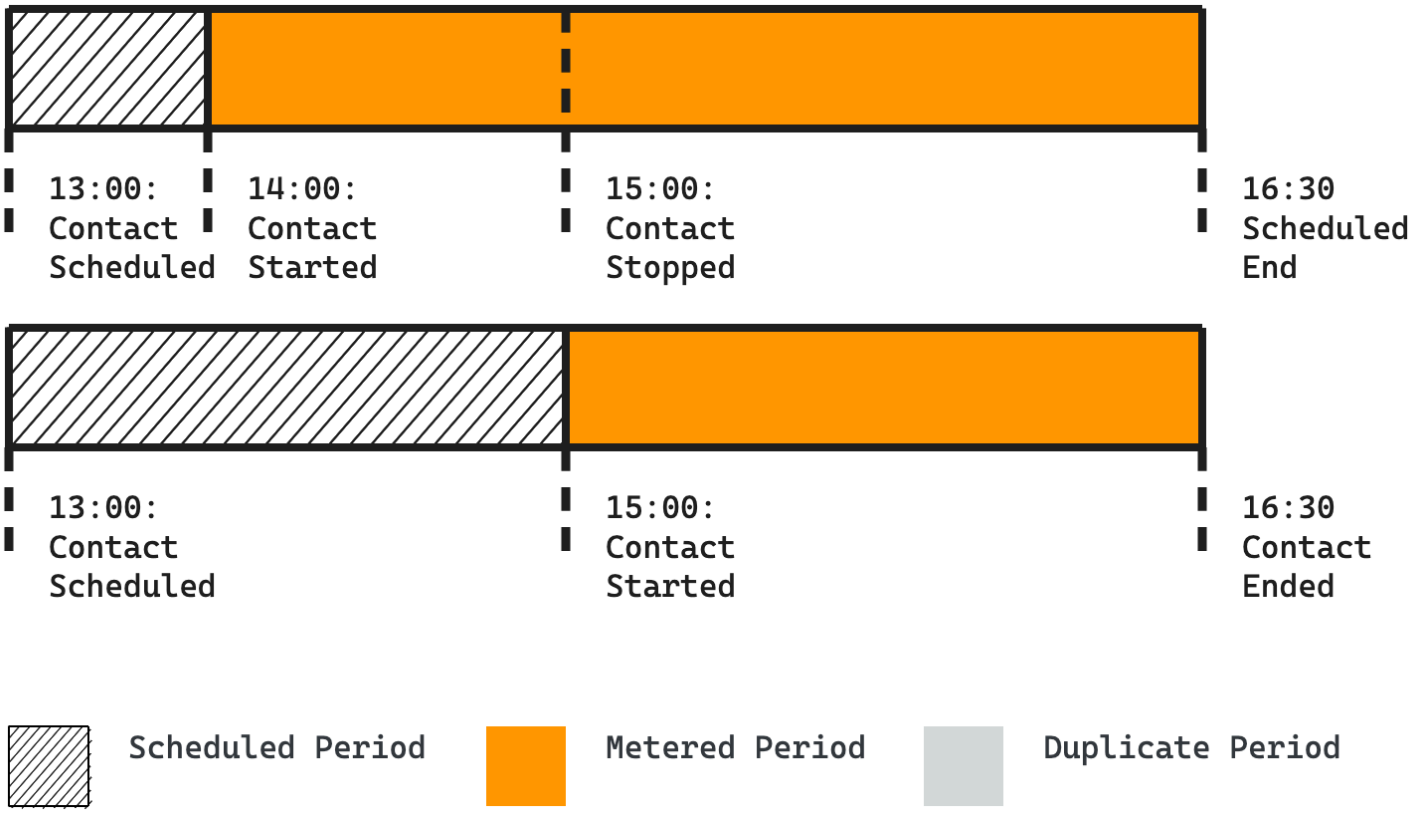
Rincian penagihan:

- Kontak pertama: 70 menit (60 menit dieksekusi +10 menit downtime sebelum kontak kedua dimulai)
- Kontak kedua: 80 menit (durasi asli penuh)

Kontak kedua ditagih untuk durasi 80 menit penuh karena Anda menghentikannya pada 15:30, meninggalkan 60 menit dari waktu yang dijadwalkan semula (15:30-16:30) tidak terisi. Kecuali Anda menjadwalkan kontak duplikat lain untuk menutupi sisa waktu, Anda bertanggung jawab atas seluruh durasi kontak yang dihentikan.

Skenario 7: stasiun Multi-antenna bumi tanpa duplikat

Pukul 13:00, Anda menjadwalkan dua kontak di Ground Station Anytown 1. Yang pertama adalah kontak 150 menit dimulai pukul 14:00 dan berakhir pada 16:30. Yang kedua adalah kontak 90 menit dimulai pukul 15:00 dan berakhir pada 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Ground Station Anytown 1 adalah stasiun tanah multi-antena, yang memungkinkan kedua kontak berjalan secara bersamaan.



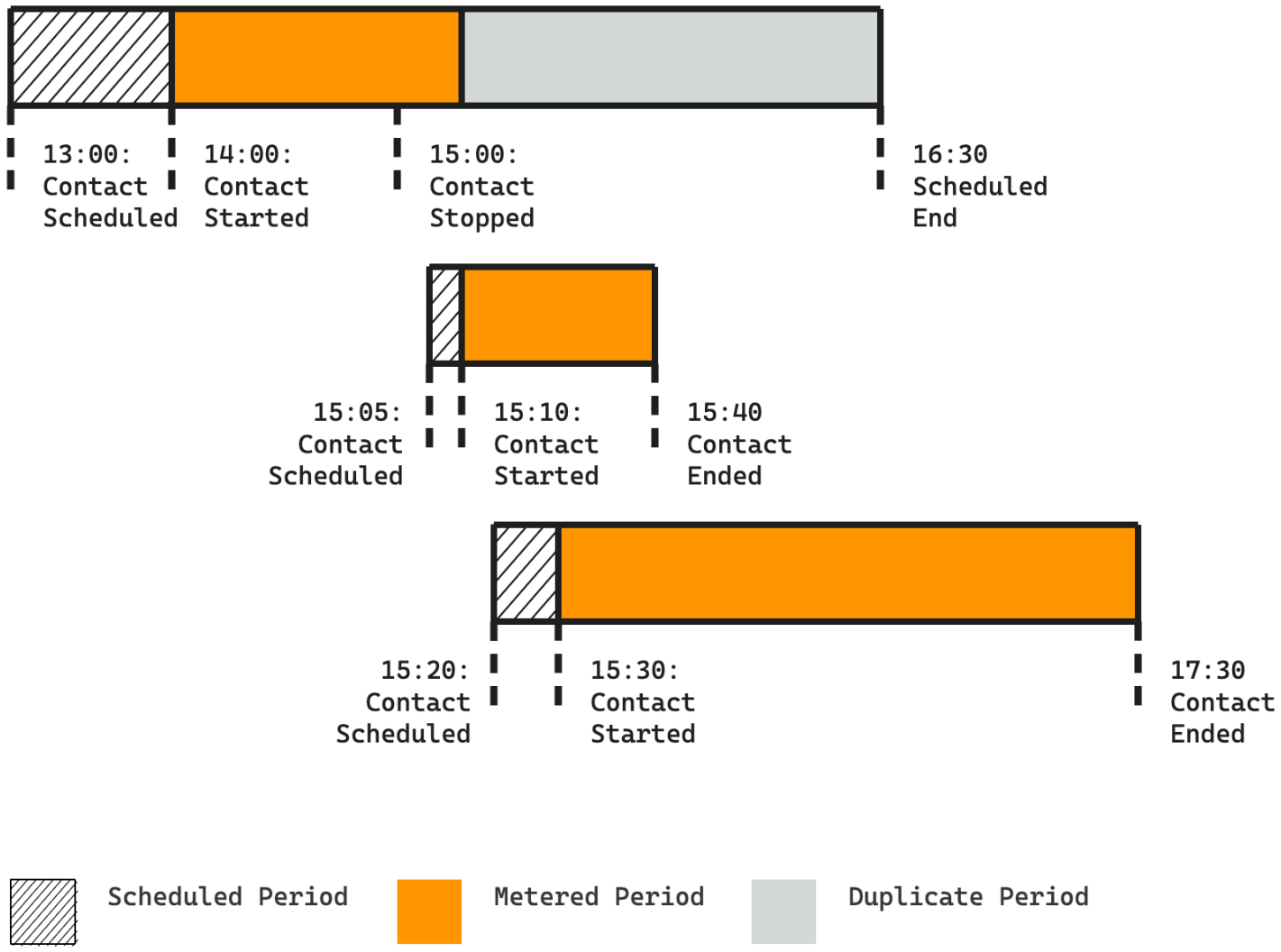
Rincian penagihan:

- Kontak pertama: 150 menit (durasi asli penuh)
- Kontak kedua: 90 menit (durasi penuh)

Meskipun kontak kedua tumpang tindih dengan bagian berhenti dari kontak pertama, itu tidak dihitung sebagai duplikat. Kontak kedua gagal memenuhi kriteria pertama untuk duplikat: dijadwalkan pada 13:00, sebelum Anda menghentikan kontak pertama pada 15:00. Karena ini bukan duplikat, Anda ditagih untuk durasi asli penuh dari kontak pertama, terlepas dari kapan Anda menghentikannya.

Skenario 8: stasiun Multi-antenna bumi dengan kontak duplikat

Anda menjadwalkan kontak 150 menit di Ground Station Anytown 1 mulai pukul 14:00 dan berakhir pukul 16:30. Pada pukul 15:00, Anda memanggil CancelContact API untuk menghentikan kontak pertama Anda. Setelah menelepon CancelContact, Anda menjadwalkan kontak 30 menit di Ground Station Anytown 1 mulai pukul 15:10 dan berakhir pukul 15:40. Kemudian, Anda menjadwalkan kontak 90 menit lagi di Ground Station Anytown 1 mulai pukul 15:30 dan berakhir pukul 17:00. Ground Station Anytown 1 adalah stasiun bumi multi-antena, yang memungkinkan kedua kontak duplikat berjalan bersamaan dengan waktu yang tumpang tindih.



Rincian penagihan:

- Kontak pertama: 70 menit (60 menit dieksekusi +10 menit downtime sebelum kontak kedua dimulai)

- Kontak kedua: 30 menit (durasi penuh)
- Kontak ketiga: 90 menit (durasi penuh)

Kedua kontak kedua dan ketiga dihitung sebagai duplikat karena Anda menjadwalkannya setelah menghentikan kontak pertama. Jarak 10 menit antara menghentikan kontak pertama (15:00) dan memulai kontak kedua (15:10) mewakili waktu henti yang ditagih terhadap kontak asli.

Perbarui kontak dan versi kontak

AWS Ground Station mendukung pembaruan kontak dengan`SCHEDULED`,`PREPASS`, atau status `PASS` kontak. Anda dapat menggunakan UpdateContactAPI untuk menentukan penggantian ephemeris untuk kontak — termasuk azimuth/elevation, data pelacakan OEM, atau TLE — tanpa membatalkan dan menjadwalkannya kembali. Ini berguna untuk operasi satelit geosynchronous (GEO) di mana Anda perlu menugaskan ulang antena ke satelit yang berbeda selama kontak, atau untuk Peluncuran dan Operasi Awal (LEOP) di mana penyesuaian penunjuk diperlukan.

Setiap kali Anda memesan atau memperbarui kontak,AWS Ground Station buat versi kontak baru. Versi kontak memberikan riwayat perubahan yang dilakukan pada kontak dan memungkinkan Anda melacak status setiap pembaruan.

Cara kerja pembuatan versi kontak

Ketika Anda memanggil ReserveContact,AWS Ground Station membuat versi pertama dari kontak (versi 1) dan mengembalikan `versionId` dalam respon. Setiap panggilan berikutnya untuk UpdateContactmembuat versi baru dengan nomor versi tambahan.

DescribeContactAPI mengembalikan versi `ACTIVE` kontak saat ini, termasuk informasi versi di `version` bidang respons. ListContactsAPI juga menyertakan informasi versi untuk setiap kontak.

Untuk melihat versi kontak tertentu, gunakan DescribeContactVersionAPI. Untuk mencantumkan semua versi kontak, gunakan ListContactVersionsAPI.

Memperbarui kontak

Anda dapat menelepon UpdateContactsaat kontak berada dalam status TERJADWAL, PREPASS, atau LULUS. API menerima parameter berikut:

- Contactid — Pengenal kontak yang akan diperbarui.

- ClientToken — Token idempotensi yang memastikan permintaan diproses hanya sekali. Jika Anda mencoba lagi permintaan dengan token klien yang sama,AWS Ground Station mengembalikan respons asli tanpa melakukan pembaruan lagi. Banyak AWS SDK secara otomatis menghasilkan token klien untuk Anda jika tidak disediakan.
- TrackingOverrides - Konfigurasi pelacakan baru untuk kontak. Ini termasuk pengaturan trek program (azimuth/elevation, TLE, atau OEM ephemeris).
- SatellitEarn — ARN satelit untuk kontak. Saat mengubah satelit target bersama dengan pengaturan trek program, berikan ARN satelit baru. Hanya pelanggan yang disetujui untuk sudut azimuth/elevation penunjuk yang dapat mengatur nilai ini ke nol. Semua pelanggan lain harus menyertakan ARN satelit kontak.

> **Important**
>
> `UpdateContact`API menerapkan semua parameter dalam permintaan. Parameter apa pun yang dihilangkan atau secara eksplisit disetel ke null diperlakukan sebagai permintaan untuk menghapus nilai itu, bukan untuk membiarkannya tidak berubah. Misalnya, jika Anda memberikan `trackingOverrides` tetapi menghilangkan`satelliteArn`, ARN satelit dihapus. Pastikan untuk menyertakan semua nilai yang diinginkan dalam setiap permintaan pembaruan.

Anda dapat mengubah satelit target selama kontak dengan memberikan yang baru `satelliteArn` bersama dengan yang sesuai`trackingOverrides`. Satelit baru harus terlihat dari stasiun bumi selama durasi kontak, karena waktu mulai dan akhir kontak tidak berubah dengan API ini. Satelit baru juga harus diangkut ke stasiun bumi dan memiliki lisensi yang diperlukan oleh profil misi. Profil misi kontak tidak dapat diubah, jadi peralihan satelit hanya berlaku ketika kedua satelit menggunakan profil misi yang sama.

> **Important**
>
> `UpdateContact`API tidak mendukung perubahan waktu mulai, waktu akhir, atau profil misi kontak. Untuk mengubah nilai-nilai ini, batalkan kontak dan pesan yang baru. `UpdateContact`API dirancang untuk menugaskan ulang konfigurasi penunjuk antena, seperti beralih antar satelit atau memperbarui data ephemeris.

> **Important**
>
> `UpdateContact`API tidak mendukung kontak yang memiliki profil misi yang menggunakan Antena Downlink Demod Decode Config konfigurasi. Untuk mengubah konfigurasi kontak ini, batalkan kontak dan pesan yang baru.

`UpdateContact`API mengembalikan `contactId` dan yang baru`versionId`. Pembaruan diproses secara asinkron. Gunakan DescribeContactVersionuntuk memeriksa status pembaruan. Beberapa AWS SDK dan AWS Command Line Interface menyediakan `ContactUpdated` pelayan yang melakukan polling hingga versi mencapai status ACTIVE atau FAILED_TO_UPDATE.

> **Note**
>
> Hanya satu pembaruan yang dapat berlangsung dalam satu waktu. Jika versi kontak terbaru dalam status UPDATE, API akan mengembalikan file`ConflictException`. Tunggu pembaruan saat ini mencapai status ACTIVE atau FAILED_TO_UPDATE sebelum mengirimkan pembaruan lain.

Status versi kontak

Setiap versi kontak memiliki salah satu status berikut:

Status	Deskripsi
UPDATING	Versi sedang diterapkan ke kontak. Pembaruan telah dikirimkan dan sedang diproses oleh AWS Ground Station.
AKTIF	Versi ini adalah konfigurasi yang saat ini aktif untuk kontak. Stasiun bumi menggunakan pengaturan versi ini.
DIGANTIKAN	Versi ini sebelumnya aktif tetapi telah digantikan oleh versi yang lebih baru.
FAILED_TO_UPDATE	Pembaruan tidak dapat diterapkan. Kontak kembali ke versi yang sebelumnya aktif. Periksa `failureCodes` dan `failureMessage` bidang untuk detailnya.

Contoh kode

Contoh berikut menunjukkan cara menggunakan API versi kontak dengan AWS SDK for Python (Boto3).

Contoh: Perbarui kontak

Contoh berikut memperbarui kontak dengan penggantian pelacakan baru dan menunggu pembaruan selesai menggunakan pelayan Boto3. `ContactUpdated`

```
import boto3
import uuid

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The contact ID of an existing scheduled contact to update
contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"

# Generate a unique client token for idempotency.
# If you retry the same request with the same client token,
# the API returns the same response without creating a duplicate version.
client_token = str(uuid.uuid4())

# Update the contact to use a different TLE ephemeris for tracking.
# The UpdateContact API applies all parameters in the request.
# Any parameter set to null is treated as a request to clear that value,
# not to leave it unchanged. Include all desired values in each request.
print(f"Updating contact {contact_id}...")

update_response = ground_station_client.update_contact(
    contactId=contact_id,
    clientToken=client_token,
    satelliteArn="arn:aws:groundstation::111122223333:satellite/a88611b0-f755-404e-
b60d-57d8aEXAMPLE",
    trackingOverrides={
        "programTrackSettings": {
            "tle": {"ephemerisId": "b2c3d4e5-6789-01ab-cdef-EXAMPLE22222"}
        }
    },
)

contact_id = update_response["contactId"]
```

```
version_id = update_response["versionId"]
print(f"Update submitted. Contact: {contact_id}, Version: {version_id}")

# Wait for the update to complete using the ContactUpdated waiter.
# The waiter polls DescribeContactVersion until the version reaches
# ACTIVE (success) or FAILED_TO_UPDATE (failure) status.
# The waiter raises WaiterError if the version reaches FAILED_TO_UPDATE
# or if MaxAttempts is exceeded, so we use try/except to handle both cases.
print("Waiting for update to complete...")

from botocore.exceptions import WaiterError

waiter = ground_station_client.get_waiter("contact_updated")

try:
    waiter.wait(
        contactId=contact_id,
        versionId=version_id,
        WaiterConfig={
            "Delay": 5,
            "MaxAttempts": 180,
        },
    )
    print(f"Contact updated successfully. Version {version_id} is now active.")
except WaiterError as e:
    # WaiterError is raised when the version reaches FAILED_TO_UPDATE
    # or when MaxAttempts is exceeded. Retrieve the current version to inspect.
    version_response = ground_station_client.describe_contact_version(
        contactId=contact_id,
        versionId=version_id,
    )
    version_status = version_response["version"]["status"]
    if version_status == "FAILED_TO_UPDATE":
        failure_codes = version_response["version"].get("failureCodes", [])
        failure_message = version_response["version"].get("failureMessage", "")
        print(f"Update failed. Codes: {failure_codes}, Message: {failure_message}")
    else:
        print(f"Waiter timed out. Current version status: {version_status}. Error:
 {e}")
```

Contoh: Jelaskan versi kontak

Contoh berikut mengambil rincian dari versi kontak tertentu, termasuk status, konfigurasi, dan informasi kegagalan.

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
version_id = 2

# Describe a specific version of a contact.
# Use this API to check the status of an update or to view the
# configuration that was active at a specific point in time.
print(f"Describing version {version_id} of contact {contact_id}...")

response = ground_station_client.describe_contact_version(
    contactId=contact_id,
    versionId=version_id,
)

# Display version details
version = response["version"]
print(f"Version ID: {version['versionId']}")
print(f"Status: {version['status']}")
print(f"Created: {version.get('created', 'N/A')}")

if version.get("activated"):
    print(f"Activated: {version['activated']}")

if version.get("superseded"):
    print(f"Superseded: {version['superseded']}")

# Display contact details for this version
print(f"\nContact ID: {response['contactId']}")
print(f"Contact Status: {response['contactStatus']}")
print(f"Ground Station: {response['groundStation']}")
print(f"Start Time: {response['startTime']}")
print(f"End Time: {response['endTime']}")

if response.get("satelliteArn"):
```

```
    print(f"Satellite ARN: {response['satelliteArn']}")

if response.get("trackingOverrides"):
    print(f"Tracking Overrides: {response['trackingOverrides']}")

# Check for failure details if the version failed to update
if version["status"] == "FAILED_TO_UPDATE":
    failure_codes = version.get("failureCodes", [])
    failure_message = version.get("failureMessage", "")
    print(f"\nFailure Codes: {failure_codes}")
    print(f"Failure Message: {failure_message}")
```

Contoh: Daftar versi kontak

Contoh berikut mencantumkan semua versi kontak untuk melihat riwayat lengkap perubahan, menggunakan pagination untuk menangani set hasil besar.

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"

# List all versions of a contact to view the full history of changes.
# Results are paginated. Use the nextToken to retrieve additional pages.
print(f"Listing versions for contact {contact_id}...")

paginator = ground_station_client.get_paginator("list_contact_versions")
page_iterator = paginator.paginate(
    contactId=contact_id,
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for version in page["contactVersionsList"]:
        version_id = version["versionId"]
        status = version["status"]
        created = version.get("created", "N/A")
```

```
        print(f"  Version {version_id}: status={status}, created={created}")

        if version.get("activated"):
            print(f"    Activated: {version['activated']}")

        if version.get("superseded"):
            print(f"    Superseded: {version['superseded']}")

        if status == "FAILED_TO_UPDATE":
            failure_codes = version.get("failureCodes", [])
            failure_message = version.get("failureMessage", "")
            print(f"    Failure Codes: {failure_codes}")
            print(f"    Failure Message: {failure_message}")

        if status == "UPDATING":
            print(f"    Update is currently in progress.")
```

Pertimbangan-pertimbangan

- Kontak yang dibuat sebelum fitur versi kontak diperkenalkan tidak memiliki informasi versi. Memanggil DescribeContactVersionatau ListContactVersionsuntuk kontak ini mengembalikan file`ResourceNotFoundException`.
- Saat Anda memperbarui penggantian pelacakan kontak yang sedang berlangsung, ada periode transisi singkat sementara antena menyesuaikan dengan konfigurasi penunjuk yang baru. Selama waktu ini, penerimaan sinyal atau transmisi dapat terganggu.
- Versi kontak tidak dapat dihapus. Gunakan ListContactVersionsuntuk melihat riwayat lengkap perubahan yang dilakukan pada kontak.
- `UpdateContact`API hanya dapat dipanggil dari wilayah penjadwalan kontak.

Gunakan AWS Ground Station fitur kembar digital

Fitur kembar digital untuk AWS Ground Station memberi Anda lingkungan di mana Anda dapat menguji dan mengintegrasikan manajemen misi satelit dan perangkat lunak perintah dan kontrol Anda. Fitur kembar digital memungkinkan Anda untuk menguji penjadwalan, verifikasi konfigurasi, dan penanganan kesalahan yang tepat tanpa menggunakan kapasitas antena produksi. Menguji AWS Ground Station integrasi Anda dengan fitur kembar digital memungkinkan Anda meningkatkan kepercayaan pada kemampuan sistem Anda untuk mengelola operasi satelit Anda dengan lancar. Ini juga memungkinkan Anda untuk menguji AWS Ground Station API tanpa menggunakan kapasitas produksi atau memerlukan lisensi spektrum.

Untuk memulai, ikutiSatelit onboard, meminta untuk masuk ke fitur kembar digital. Setelah satelit Anda terhubung ke fitur kembar digital, Anda dapat menjadwalkan kontak dengan stasiun darat kembar digital. Daftar stasiun bumi yang dapat Anda akses dapat diambil melalui respons AWS SDK ListGroundStations. Stasiun ground kembar digital adalah salinan persis dari stasiun bumi yang terdaftar AWS Ground Station Lokasi dengan awalan modifikasi ke Ground Station Nama “Digital Twin”. Ini termasuk kemampuan antena dan metadata mereka, termasuk, namun tidak terbatas pada, masker situs dan koordinat GPS yang sebenarnya. Saat ini, fitur kembar digital tidak mendukung pengiriman data seperti yang dijelaskan dalamBekerja dengan aliran data. Selain itu, pengiriman telemetri seperti yang dijelaskan dalam Bekerja dengan telemetri tidak didukung pada stasiun bumi kembar digital.

Setelah onboard, fitur kembar digital memancarkan EventBridge peristiwa Amazon dan respons API yang sama seperti layanan produksi seperti yang dijelaskan dalam. Otomatisasi AWS Ground Station dengan Acara Peristiwa ini akan memungkinkan Anda untuk menyempurnakan konfigurasi dan grup titik akhir aliran data Anda.

AWS Ground Station Antena Khusus

AWS Ground Station Antena Khusus adalah sistem antena yang dibuat khusus yang AWS mengelola atas nama Anda. Berbeda dengan AWS Ground Station antena publik tempat Anda berbagi waktu antena dengan pelanggan lain, Antena Khusus memberi Anda akses khusus ke antena yang dibangun sesuai spesifikasi Anda. Anda dapat memiliki satu atau lebih Antena Khusus. Antena Khusus terhubung ke AWS jaringan global, dan Anda berinteraksi dengannya menggunakan alur kerja yang sama AWS Ground Station APIs dan yang Anda gunakan dengan antena publik, dengan penambahan visibilitas yang ditingkatkan ke pemanfaatan antena.

Apa itu Antena Khusus

Antena Khusus adalah sistem antena fisik yang dibuat khusus untuk organisasi Anda dan dikelola sepenuhnya oleh. AWS Anda dapat memiliki satu atau lebih Antena Khusus. Antena Khusus berbeda dari AWS Ground Station antena publik dengan cara berikut:

- Dibuat khusus - Antena Khusus dibangun sesuai spesifikasi Anda. Kemampuan Antena Khusus tidak terbatas pada kemampuan antena publik seperti yang dijelaskan dalamAWS Ground Station Kemampuan Situs.
- Lokasi fleksibel - Antena Khusus tidak terbatas pada lokasi AWS Ground Station antena yang ada. Antena Khusus dapat dihubungkan ke AWS Wilayah mana pun yang ada, termasuk wilayah yang saat AWS Ground Station ini tidak tersedia. Anda dapat bekerja dengan AWS untuk menentukan lokasi dan wilayah yang memenuhi kebutuhan Anda.
- Akses khusus — Anda memiliki akses khusus ke antena daripada berbagi waktu antena dengan AWS Ground Station pelanggan lain berdasarkan per kontak.

Semua AWS Ground Station antena, termasuk Antena Khusus, dikelola sepenuhnya oleh. AWS Ini termasuk pemeliharaan dan konektivitas ke AWS jaringan global. Anda berinteraksi dengan Antena Khusus menggunakan alur kerja yang sama AWS Ground Station APIs dan yang Anda gunakan dengan antena publik. Anda menjadwalkan kontak, mengonfigurasi profil misi, dan mengirimkan data dengan cara yang sama.

Antena Khusus dapat dibagikan oleh beberapa AWS akun, di mana pelanggan yang memegang kontrak Antena Khusus memilih akun mana yang akan dionboard. Setiap akun onboard dapat menjadwalkan kontak pada antena secara independen, dan memiliki visibilitas ke reservasi di semua akun yang berbagi antena.

Untuk mempelajari lebih lanjut tentang Antena Khusus atau untuk memulai, hubungi AWS Dukungan melalui. AWS Support Center Console

Peningkatan visibilitas reservasi

Saat Anda menggunakan ListGroundStationReservationsAPI terhadap Antena Khusus, Anda melihat informasi tambahan yang tidak tersedia di antena publik. Tabel berikut merangkum perbedaan perilaku ListGroundStationReservationsAPI untuk Antena Khusus dibandingkan dengan antena publik AWS Ground Station .

Perilaku	Antena Khusus	Antena publik
Kontak Anda sendiri	Terlihat dengan detail lengkap, termasuk `contactId`	Terlihat dengan detail lengkap, termasuk `contactId`
Kontak akun lain	Terlihat dengan slot waktu saja, tanpa `contactId`	Tidak terlihat
Jendela pemeliharaan	Terlihat dengan `PLANNED` atau `UNPLANNED` `maintenanceType`	Tidak terlihat

Jendela pemeliharaan mewakili periode ketika antena tidak tersedia untuk komunikasi satelit. `maintenanceType`Bidang menunjukkan apakah pemeliharaan itu `PLANNED` atau`UNPLANNED`. Ketika pemeliharaan yang tidak direncanakan dijadwalkan, kontak yang tumpang tindih dengan jendela pemeliharaan dapat dibatalkan oleh. AWS Ground Station

Ketika Anda melihat kontak dari AWS akun lain yang berbagi Antena Khusus Anda, reservasi mencakup slot waktu dan informasi antena, tetapi `contactId` tidak termasuk.

> **Important**
>
> Visibilitas reservasi yang ditingkatkan hanya berlaku untuk Antena Khusus Anda. Ketika Anda menggunakan AWS Ground Station antena publik, Anda memiliki visibilitas yang sama seperti pelanggan lainnya. Anda tidak melihat jendela pemeliharaan atau reservasi dari akun lain di antena publik.

Untuk informasi selengkapnya tentang reservasi cantuman, lihatLihat reservasi stasiun bumi.

Sumber daya terkait

- Lihat reservasi stasiun bumi
- AWS Ground Station Lokasi
- AWS Ground Station Kemampuan Situs
- ListGroundStationReservations di Referensi API AWS Ground Station

Memahami pemantauan dengan AWS Ground Station

Pemantauan merupakan bagian penting dari menjaga keandalan, ketersediaan, dan kinerja AWS Ground Station. AWS menyediakan alat pemantauan berikut untuk menonton AWS Ground Station, melaporkan ketika ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu.

- Amazon EventBridge Events memberikan aliran peristiwa sistem yang mendekati real-time yang menjelaskan perubahan AWS sumber daya. EventBridge Peristiwa memungkinkan komputasi berbasis peristiwa otomatis, karena Anda dapat menulis aturan yang mengawasi peristiwa tertentu dan memicu tindakan otomatis di AWS layanan lain saat peristiwa ini terjadi. Untuk informasi selengkapnya tentang EventBridge Acara, lihat Panduan Pengguna EventBridge Acara Amazon.
- AWS CloudTrailmenangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama AWS akun Anda dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang dipanggil AWS, alamat IP sumber dari mana panggilan dilakukan, dan kapan panggilan terjadi. Untuk informasi selengkapnya AWS CloudTrail, lihat Panduan AWS CloudTrail Pengguna.
- Amazon CloudWatch Metrics menangkap metrik untuk kontak terjadwal Anda saat menggunakan. AWS Ground Station CloudWatch Metrik memungkinkan Anda menganalisis data berdasarkan saluran, polarisasi, dan ID satelit untuk mengidentifikasi kekuatan dan kesalahan sinyal dalam kontak Anda. Untuk informasi selengkapnya, lihat Menggunakan CloudWatch metrik Amazon.
- AWS Notifikasi Pengguna dapat digunakan untuk menyiapkan saluran pengiriman agar mendapat pemberitahuan tentang AWS Ground Station peristiwa. Anda akan menerima notifikasi saat ada sebuah peristiwa yang cocok dengan sebuah aturan yang Anda tentukan. Anda dapat menerima pemberitahuan untuk acara melalui beberapa saluran, termasuk email, Pengembang Amazon Q dalam pemberitahuan obrolan aplikasi obrolan, atau pemberitahuan AWS Console Mobile Applicationpush. Anda juga dapat melihat notifikasi di pusat Pemberitahuan AWS Konsol. Notifikasi Pengguna mendukung agregasi, yang dapat mengurangi jumlah notifikasi yang Anda terima selama acara tertentu.

Gunakan topik berikut untuk memantau AWS Ground Station.

Topik

- Otomatisasi AWS Ground Station dengan Acara
- Log panggilan AWS Ground Station API dengan AWS CloudTrail
- Lihat metrik dengan Amazon CloudWatch

Otomatisasi AWS Ground Station dengan Acara

> **Note**
>
> Dokumen ini menggunakan istilah “acara” di seluruh. CloudWatch Peristiwa dan EventBridge merupakan layanan dan API dasar yang sama. Aturan untuk mencocokkan peristiwa yang masuk dan meruteknya ke target untuk diproses dapat dibuat menggunakan salah satu layanan.

Acara memungkinkan Anda untuk mengotomatiskan AWS layanan Anda dan merespons secara otomatis peristiwa sistem seperti masalah ketersediaan aplikasi atau perubahan sumber daya. Acara dari AWS layanan disampaikan dalam waktu dekat. Anda dapat menulis aturan sederhana untuk menunjukkan kejadian mana yang sesuai kepentingan Anda, dan tindakan otomatis apa yang diambil ketika suatu kejadian sesuai dengan suatu aturan. Beberapa tindakan yang dapat dipicu secara otomatis termasuk yang berikut:

- Memanggil fungsi AWS Lambda
- Meminta Perintah Amazon EC2 Run
- Mengirim peristiwa ke Amazon Kinesis Data Streams
- Mengaktifkan mesin AWS Step Functions negara
- Memberi tahu topik Amazon SNS atau antrean Amazon SQS

Beberapa contoh penggunaan acara dengan AWS Ground Station meliputi:

- Memanggil fungsi Lambda untuk mengotomatiskan awal dan penghentian instans Amazon EC2 berdasarkan status peristiwa.
- Menerbitkan ke topik Amazon SNS setiap kali kontak berubah status. Topik-topik ini dapat diatur untuk mengirimkan pemberitahuan email di awal atau akhir kontak.

Untuk informasi selengkapnya, lihat Panduan Pengguna EventBridge Acara Amazon.

AWS Ground Station Jenis Acara

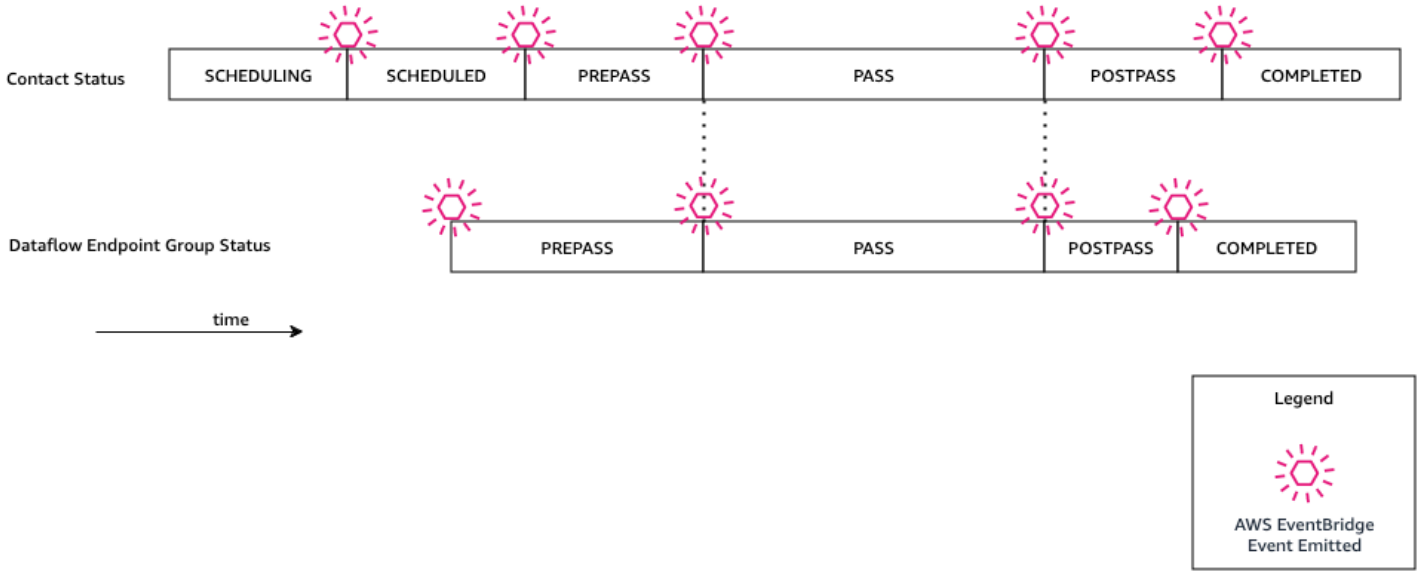
> **Note**
>
> Semua peristiwa yang dihasilkan oleh AWS Ground Station memiliki “aws.groundstation” sebagai nilai untuk “sumber”.

AWS Ground Station memancarkan peristiwa yang terkait dengan perubahan status untuk mendukung kemampuan Anda menyesuaikan otomatisasi Anda. Saat ini, AWS Ground Station mendukung peristiwa perubahan status kontak, peristiwa perubahan grup titik akhir aliran data, dan peristiwa perubahan status ephemeris. Bagian berikut memberikan informasi rinci tentang setiap jenis.

Hubungi Timeline Acara

AWS Ground Station memancarkan peristiwa saat kontak Anda mengubah status. Untuk informasi lebih lanjut tentang apa perubahan keadaan itu, dan apa arti negara itu sendiri, lihatMemahami siklus hidup kontak. Setiap grup titik akhir aliran data yang digunakan dalam kontak Anda memiliki serangkaian peristiwa independen yang juga dipancarkan. Selama jangka waktu yang sama, kami juga memancarkan peristiwa untuk grup titik akhir aliran data Anda. Waktu yang tepat dari peristiwa pra-lulus dan pasca-lulus dapat dikonfigurasi oleh Anda saat Anda mengatur profil misi dan grup titik akhir aliran data Anda.

Diagram berikut menunjukkan status dan peristiwa yang dipancarkan untuk kontak nominal dan kelompok titik akhir aliran data terkait.



Perubahan Status Kontak Ground Station

Jika Anda ingin melakukan tindakan tertentu saat kontak yang akan datang mengubah status, Anda dapat mengatur aturan untuk mengotomatiskan tindakan ini. Ini berguna ketika Anda ingin menerima pemberitahuan tentang perubahan status kontak Anda. Jika Anda ingin mengubah saat menerima acara ini, Anda dapat memodifikasi profil misi Anda contactPrePassDurationSecondsdan contactPostPassDurationSeconds. Acara dikirim ke wilayah tempat kontak dijadwalkan.

Contoh acara disediakan di bawah ini.

```
{
    "version": "0",
    "id": "01234567-0123-0123",
    "account": "123456789012",
    "time": "2019-05-30T17:40:30Z",
    "region": "us-west-2",
    "source": "aws.groundstation",
    "resources": [
        "arn:aws:groundstation:us-
west-2:123456789012:contact/11111111-1111-1111-1111-111111111111"
    ],
    "detailType": "Ground Station Contact State Change",
    "detail": {
        "contactId": "11111111-1111-1111-1111-111111111111",
        "groundstationId": "Ground Station 1",
```

```
        "missionProfileArn": "arn:aws:groundstation:us-west-2:123456789012:mission-
profile/11111111-1111-1111-1111-111111111111",
        "satelliteArn":
 "arn:aws:groundstation::123456789012:satellite/11111111-1111-1111-1111-111111111111",
        "contactStatus": "PASS"
    }
}
```

Nilai yang mungkin untuk `contactStatus` didefinisikan dalamthe section called “AWS Ground Station status kontak”.

Perubahan Status Grup Ground Station Dataflow Endpoint

Jika Anda ingin melakukan tindakan saat grup titik akhir aliran data Anda digunakan untuk menerima data, Anda dapat menyiapkan aturan untuk mengotomatiskan tindakan ini. Ini akan memungkinkan Anda untuk melakukan tindakan yang berbeda dalam menanggapi status perubahan status grup titik akhir dataflow. Jika Anda ingin mengubah saat menerima peristiwa ini, gunakan grup titik akhir aliran data dengan dan. contactPrePassDurationSecondscontactPostPassDurationSeconds Acara ini akan dikirim ke wilayah grup endpoint aliran data.

Contoh diberikan di bawah ini.

```
{
    "version": "0",
    "id": "01234567-0123-0123",
    "account": "123456789012",
    "time": "2019-05-30T17:40:30Z",
    "region": "us-west-2",
    "source": "aws.groundstation",
    "resources": [
        "arn:aws:groundstation:us-west-2:123456789012:dataflow-endpoint-group/
bad957a8-1d60-4c45-a92a-39febd98921d",
        "arn:aws:groundstation:us-west-2:123456789012:contact/98ddd10f-f2bc-479c-
bf7d-55644737fb09",
        "arn:aws:groundstation:us-west-2:123456789012:mission-profile/c513c84c-
eb40-4473-88a2-d482648c9234"
    ],
    "detailType": "Ground Station Dataflow Endpoint Group State Change",
    "detail": {
        "dataflowEndpointGroupId": "bad957a8-1d60-4c45-a92a-39febd98921d",
        "groundstationId": "Ground Station 1",
        "contactId": "98ddd10f-f2bc-479c-bf7d-55644737fb09",
```

```
        "dataflowEndpointGroupArn": "arn:aws:groundstation:us-
west-2:680367718957:dataflow-endpoint-group/bad957a8-1d60-4c45-a92a-39febd98921d",
        "missionProfileArn": "arn:aws:groundstation:us-west-2:123456789012:mission-
profile/c513c84c-eb40-4473-88a2-d482648c9234",
        "dataflowEndpointGroupState": "PREPASS"
    }
}
```

Kemungkinan negara untuk `dataflowEndpointGroupState` memasukkan`PREPASS`,`PASS`,`POSTPASS`, dan`COMPLETED`.

Acara Ephemeris

Ground Station Perubahan Negara Ephemeris

Jika Anda ingin melakukan tindakan saat ephemeris mengubah status, Anda dapat mengatur aturan untuk mengotomatiskan tindakan ini. Ini memungkinkan Anda untuk melakukan tindakan yang berbeda sebagai respons terhadap keadaan perubahan ephemeris. Misalnya, Anda dapat melakukan tindakan ketika ephemeris telah menyelesaikan validasi, dan sekarang. `ENABLED` Pemberitahuan untuk acara ini akan dikirim ke wilayah jika ephemeris diunggah.

Contoh diberikan di bawah ini.

```
{
    "id": "7bf73129-1428-4cd3-a780-95db273d1602",
    "detail-type": "Ground Station Ephemeris State Change",
    "source": "aws.groundstation",
    "account": "123456789012",
    "time": "2019-12-03T21:29:54Z",
    "region": "us-west-2",
    "resources": [
        "arn:aws:groundstation::123456789012:satellite/10313191-c9d9-4ecb-a5f2-
bc55cab050ec",
        "arn:aws:groundstation::123456789012:ephemeris/111111-cccc-bbbb-a555-
bcccca005000"
    ],
    "detail": {
        "ephemerisStatus": "ENABLED",
        "ephemerisId": "111111-cccc-bbbb-a555-bcccca005000",
        "satelliteId": "10313191-c9d9-4ecb-a5f2-bc55cab050ec"
    }
```

```
}
```

Kemungkinan negara untuk `ephemerisStatus` memasukkan`ENABLED`,`VALIDATING`,`INVALID`,`ERROR`,`DISABLED`, `EXPIRED`

Log panggilan AWS Ground Station API dengan AWS CloudTrail

AWS Ground Station terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Ground Station. CloudTrail menangkap semua panggilan API untuk AWS Ground Station sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari AWS Ground Station konsol dan panggilan kode ke operasi AWS Ground Station API. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk. AWS Ground Station Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat AWS Ground Station, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

AWS Ground Station Informasi di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Ketika aktivitas terjadi di AWS Ground Station, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat Melihat Acara dengan Riwayat CloudTrail Acara.

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk AWS Ground Station, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol tersebut, jejak tersebut diterapkan ke semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- Gambaran Umum untuk Membuat Jejak
- CloudTrail Layanan dan Integrasi yang Didukung

- Mengonfigurasi Notifikasi Amazon SNS untuk CloudTrail
- Menerima File CloudTrail Log dari Beberapa Wilayah dan Menerima File CloudTrail Log dari Beberapa Akun

Semua AWS Ground Station tindakan dicatat oleh CloudTrail dan didokumentasikan dalam Referensi AWS Ground Station API. Misalnya, panggilan ke`ReserveContact`, `CancelContact` dan `ListConfigs` tindakan menghasilkan entri dalam file CloudTrail log.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut ini:

- Apakah permintaan itu dibuat dengan kredenal pengguna root atau AWS Identity and Access Management (IAM).
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi lain, lihat Elemen userIdentity CloudTrail .

Memahami Entri File AWS Ground Station Log

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber mana pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ReserveContact` tindakan.

Contoh: ReserveContact

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:sts::123456789012:user/Alice",
```

```
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2019-05-15T21:11:59Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::123456789012:role/Alice",
                "accountId": "123456789012",
                "userName": "Alice"
            }
        }
    },
    "eventTime": "2019-05-15T21:14:37Z",
    "eventSource": "groundstation.amazonaws.com",
    "eventName": "ReserveContact",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "Mozilla/5.0 Gecko/20100101 Firefox/123.0",
    "requestParameters": {
        "satelliteArn":
 "arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555",
        "groundStation": "Ohio 1",
        "startTime": 1558356107,
        "missionProfileArn": "arn:aws:groundstation:us-east-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555",
        "endTime": 1558356886
    },
    "responseElements": {
        "contactId": "11111111-2222-3333-4444-555555555555"
    },
    "requestID": "11111111-2222-3333-4444-555555555555",
    "eventID": "11111111-2222-3333-4444-555555555555",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "11111111-2222-3333-4444-555555555555"
}
```

Lihat metrik dengan Amazon CloudWatch

Selama kontak, AWS Ground Station secara otomatis menangkap dan mengirim data CloudWatch untuk analisis. Data Anda dapat dilihat di CloudWatch konsol Amazon. Untuk informasi selengkapnya tentang mengakses dan CloudWatch Metrik, lihat Menggunakan Metrik Amazon CloudWatch.

Fitur AWS Ground Station telemetri juga dapat digunakan untuk menerima metrik mendekati waktu nyata selama kontak. CloudWatch metrik tidak tersedia dalam waktu dekat dan mungkin mengalami keterlambatan pengiriman. CloudWatch juga menggabungkan metrik selama periode satu detik, berpotensi mengurangi granularitas data. Fitur telemetri menyediakan metrik individual dan mengirimkannya dalam waktu dekat langsung ke akun Anda. AWS Untuk informasi selengkapnya, lihat Bekerja dengan telemetri.

Important

AWS Ground Station memancarkan CloudWatch metrik ke AWS wilayah yang terkait dengan lokasi stasiun darat kontak, bukan AWS wilayah tempat kontak dijadwalkan. Untuk melihat metrik untuk kontak, Anda harus mengakses CloudWatch di wilayah stasiun bumi. Untuk informasi tentang AWS wilayah mana yang terkait dengan setiap lokasi stasiun bumi, lihatMenemukan AWS wilayah untuk lokasi stasiun bumi. Untuk menerima data telemetri di wilayah tempat Anda menjadwalkan kontak, Anda dapat menggunakan fitur AWS Ground Station telemetri. Lihat Bekerja dengan telemetri untuk detail selengkapnya.

AWS Ground Station Metrik dan Dimensi

Metrik apa yang tersedia?

Metrik berikut tersedia dari AWS Ground Station.

Note

Metrik spesifik yang dipancarkan tergantung pada AWS Ground Station kemampuan yang digunakan. Bergantung pada konfigurasi Anda, hanya sebagian dari metrik di bawah ini yang dapat dipancarkan.

Metrik	Dimensi Metrik	Deskripsi
`AzimuthAngle`	SatelliteId	Sudut azimuth antena. Utara sejati adalah 0 derajat dan timur 90 derajat. Unit: derajat
`BitErrorRate`	Saluran, Polarisasi, SatelliteId	Tingkat kesalahan pada bit dalam jumlah transmisi bit tertentu. Kesalahan bit disebabkan oleh kebisinga n, distorsi, atau gangguan Unit: Kesalahan bit per satuan waktu
`BlockErrorRate`	Saluran, Polarisasi, SatelliteId	Tingkat kesalahan blok dalam jumlah tertentu dari blok yang diterima. Kesalahan blok disebabkan oleh interferensi. Unit: Blok yang salah/Jumlah total blok

Metrik	Dimensi Metrik	Deskripsi
`CarrierFrequencyRecovery_Cn0`	Kategori, Config, SatelliteId	Rasio kepadatan pembawa terhadap kebisingan per unit bandwidth. Satuan: Desibel-Hertz (dB-Hz)
`CarrierFrequencyRecovery_Locked`	Kategori, Config, SatelliteId	Atur ke 1 saat loop pemulihan frekuensi pembawa demodulator terkunci dan 0 saat dibuka kuncinya. Unit: tanpa unit

Metrik	Dimensi Metrik	Deskripsi
`CarrierFrequencyRecovery_OffsetFrequency_Hz`	Kategori, Config, SatelliteId	Offset antara pusat sinyal yang diperkirakan dan frekuensi pusat ideal. Hal ini disebabkan oleh pergeseran Doppler dan offset osilator lokal antara pesawat ruang angkasa dan sistem antena. Satuan: hertz (Hz)
`ElevationAngle`	SatelliteId	Sudut elevasi antena. Cakrawala adalah 0 derajat dan zenith adalah 90 derajat. Unit: derajat
`Es/N0`	Saluran, Polarisasi, SatelliteId	Rasio energi per simbol terhadap kerapatan spektral daya kebisingan. Unit: desibel (dB)

Metrik	Dimensi Metrik	Deskripsi
`ReceivedPower`	Polarisasi SatelliteId	Kekuatan sinyal yang diukur dalam demodulator/decoder. Satuan: desibel relatif terhadap miliwatt (dBm)
`SymbolTimingRecovery_ErrorVectorMagnitude`	Kategori, Config, SatelliteId	Besarnya vektor kesalahan antara simbol yang diterima dan titik konstelasi ideal. Unit: persen
`SymbolTimingRecovery_Locked`	Kategori, Config, SatelliteId	Setel ke 1 saat loop pemulihan waktu simbol demodulator terkunci dan 0 saat dibuka Unit: tanpa unit

Metrik	Dimensi Metrik	Deskripsi
`SymbolTimingRecovery_OffsetSymbolRate`	Kategori, Config, SatelliteId	Offset antara perkiraan tingkat simbol dan tingkat simbol sinyal ideal. Hal ini disebabkan oleh pergeseran Doppler dan offset osilator lokal antara pesawat ruang angkasa dan sistem antena. Satuan: simbol/detik

Dimensi apa yang digunakan AWS Ground Station?

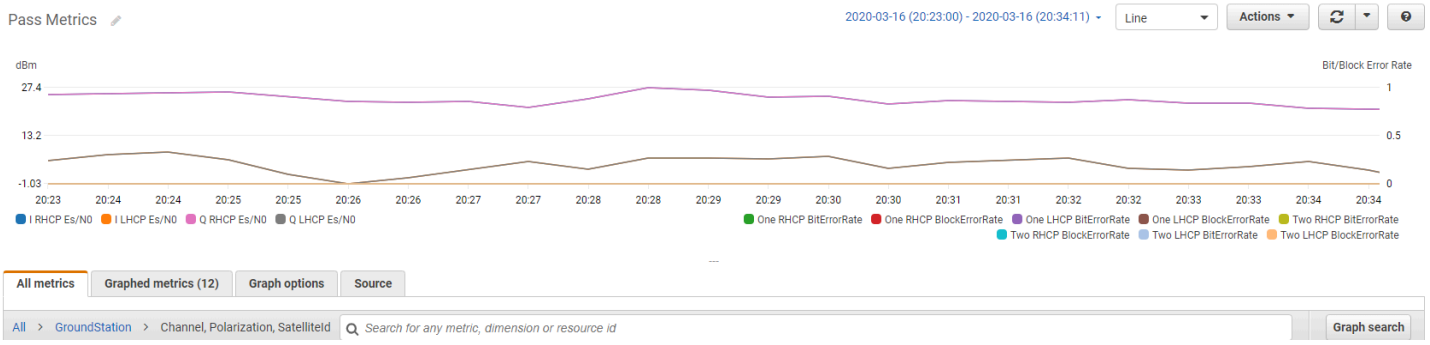
Anda dapat memfilter AWS Ground Station data menggunakan dimensi berikut.

Dimensi	Deskripsi
`Category`	Demodulasi atau Decode.
`Channel`	Saluran untuk setiap kontak termasuk Satu, Dua, I (dalam fase), dan Q (kuadratur).
`Config`	Konfigurasi decode demod downlink antena arn.
`Polarization`	Polarisasi untuk setiap kontak termasuk LHCP (Left Hand Circular Polarized) atau RHCP (Tangan Kanan Circular Polarized).
`SatelliteId`	ID satelit berisi ARN satelit untuk kontak Anda.

Melihat metrik

Saat melihat metrik grafik, penting untuk dicatat bahwa jendela agregasi menentukan bagaimana metrik Anda akan ditampilkan. Setiap metrik dalam kontak dapat ditampilkan sebagai data per detik selama 3 jam setelah data diterima. Data Anda akan dikumpulkan oleh CloudWatch Metrik sebagai data per menit setelah periode 3 jam berlalu. Jika Anda perlu melihat metrik pada pengukuran data per detik, disarankan untuk melihat data Anda dalam periode 3 jam setelah data diterima atau disimpan di luar Metrik. CloudWatch Untuk informasi selengkapnya tentang CloudWatch retensi, lihat CloudWatch Konsep Amazon - Retensi metrik.

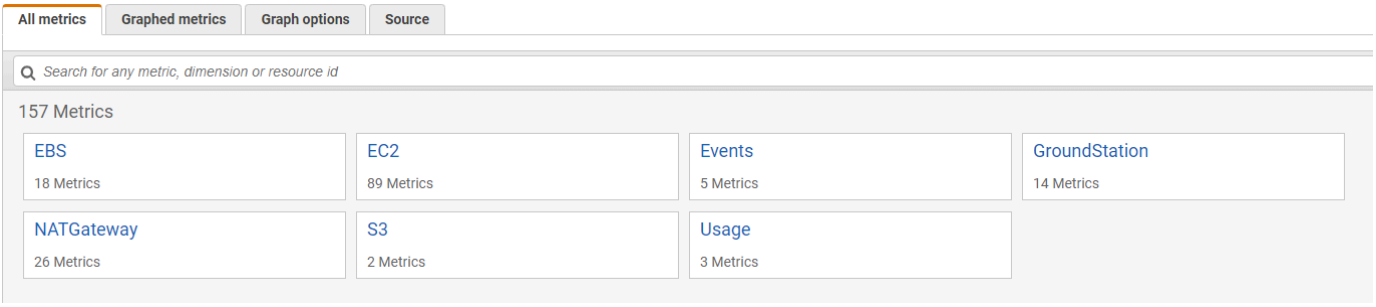
Selain itu, data apa pun yang diambil dalam 60 detik pertama tidak akan berisi informasi yang cukup untuk menghasilkan metrik yang berarti, dan kemungkinan tidak akan ditampilkan. Untuk melihat metrik yang bermakna, disarankan untuk melihat data Anda setelah 60 detik berlalu.



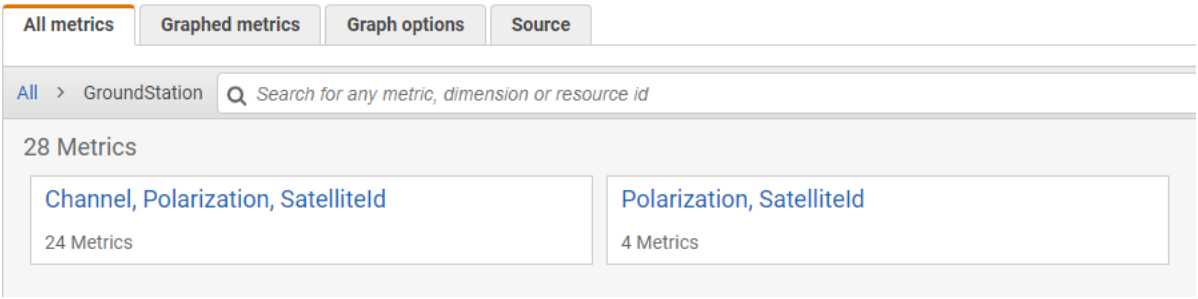
Untuk informasi selengkapnya tentang AWS Ground Station metrik grafik CloudWatch, lihat Metrik Grafik.

Untuk melihat metrik menggunakan konsol

1. Tentukan AWS wilayah yang terkait dengan lokasi stasiun bumi Anda. AWS Ground Station memancarkan CloudWatch metrik di wilayah yang terkait dengan lokasi stasiun darat kontak Anda. Untuk daftar lokasi stasiun bumi dan AWS wilayah terkaitnya, lihatMenemukan AWS wilayah untuk lokasi stasiun bumi.
2. Buka konsol CloudWatch .
3. Pada panel navigasi, silakan pilih Metrik.
4. Pilih GroundStationnamespace.



5. Pilih dimensi metrik yang Anda inginkan (misalnya, Saluran, Polarisasi, SatelliteId).



6. Tab Semua metrik menampilkan semua metrik dimensi tersebut di namespace. Anda dapat melakukan hal berikut:

   a. Untuk menyortir tabel, gunakan judul kolomnya.

   b. Untuk membuat grafik metrik, pilih kotak centang yang terkait dengan metrik. Untuk memilih semua metrik, pilih kotak centang di baris judul tabel.

   c. Untuk memfilter berdasarkan sumber daya, pilih ID sumber daya, lalu pilih Tambahkan ke pencarian.

   d. Untuk menyaring berdasarkan metrik, pilih nama metrik, kemudian pilih Tambahkan ke pencarian.

Untuk melihat metrik menggunakan AWS CLI

AWS Ground Station memancarkan CloudWatch metrik di wilayah yang terkait dengan lokasi stasiun darat kontak Anda. Untuk daftar lokasi stasiun bumi, mereka berada di AWS wilayah terkait,Menemukan AWS wilayah untuk lokasi stasiun bumi. Ganti *ground-station-region-code* dengan kode AWS wilayah untuk lokasi stasiun bumi Anda (misalnya, `us-west-2` untuk Oregon 1, Hawaii 1, atau Alaska 1). Semua AWS CLI perintah selanjutnya dalam prosedur ini harus menggunakan wilayah yang sama.

1. Pastikan AWS CLI sudah terpasang. Untuk informasi tentang penginstalan AWS CLI, lihat Menginstal AWS CLI versi 2.
2. Identifikasi AWS wilayah yang terkait dengan lokasi stasiun bumi Anda.
3. Gunakan get-metric-datametode CloudWatch CLI untuk menghasilkan file yang dapat dimodifikasi untuk menentukan metrik yang Anda minati, dan kemudian digunakan untuk menanyakan metrik tersebut.

   Untuk melakukan ini, jalankan yang berikut:`aws cloudwatch get-metric-data --region` *`ground-station-region-code`* `--generate-cli-skeleton`. Ini akan menghasilkan output yang mirip dengan:

```
{
   "MetricDataQueries": [
      {
         "Id": "",
         "MetricStat": {
            "Metric": {
               "Namespace": "",
               "MetricName": "",
               "Dimensions": [
                  {
                     "Name": "",
                     "Value": ""
                  }
               ]
            },
            "Period": 0,
            "Stat": "",
            "Unit": "Seconds"
         },
         "Expression": "",
         "Label": "",
         "ReturnData": true,
         "Period": 0,
         "AccountId": ""
      } ],
   "StartTime": "1970-01-01T00:00:00",
   "EndTime": "1970-01-01T00:00:00",
   "NextToken": "",
   "ScanBy": "TimestampDescending",
   "MaxDatapoints": 0,
```

```
            "LabelOptions": {
               "Timezone": ""
            }
         }
```

4. Buat daftar CloudWatch metrik yang tersedia dengan menjalankan`aws cloudwatch list-metrics --region` *`ground-station-region-code`*.

   Jika Anda baru saja menggunakan AWS Ground Station, metode harus mengembalikan output yang berisi entri seperti:

```
         ...
         {
            "Namespace": "AWS/GroundStation",
            "MetricName": "ReceivedPower",
            "Dimensions": [
               {
                  "Name": "Polarization",
                  "Value": "LHCP"
               },
               {
                  "Name": "SatelliteId",
                  "Value": "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-
bbbb-cccc-dddd-eeeeeeeeeeee"
               }
            ]
       },
       ...
```

   > **Note**
   >
   > Jika sudah lebih dari 2 minggu sejak terakhir kali digunakan AWS Ground Station, maka Anda perlu memeriksa tabel metrik yang tersedia secara manual untuk menemukan nama dan dimensi metrik di namespace `AWS/GroundStation` metrik. Untuk informasi selengkapnya tentang CloudWatch batasan, lihat: Melihat metrik yang tersedia

5. Ubah file JSON yang Anda buat di langkah 2 agar sesuai dengan nilai yang diperlukan dari langkah 3, misalnya`SatelliteId`, dan `Polarization` dari metrik Anda. Pastikan juga untuk memperbarui`StartTime`, dan `EndTime` nilai agar sesuai dengan kontak Anda. Contoh:

```
        {
           "MetricDataQueries": [
              {
                 "Id": "receivedPowerExample",
                 "MetricStat": {
                    "Metric": {
                       "Namespace": "AWS/GroundStation",
                       "MetricName": "ReceivedPower",
                       "Dimensions": [
                          {
                             "Name": "SatelliteId",
                             "Value":
 "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-bbbb-cccc-dddd-
eeeeeeeeeeee"
                          },
                          {
                             "Name": "Polarization",
                             "Value": "RHCP"
                          }
                       ]
                    },
                    "Period": 300,
                    "Stat": "Maximum",
                    "Unit": "None"
                 },
                 "Label": "ReceivedPowerExample",
                 "ReturnData": true
              }
           ],
           "StartTime": "2024-02-08T00:00:00",
           "EndTime": "2024-04-09T00:00:00"
        }
```

> **Note**
>
> AWS Ground Station menerbitkan metrik setiap 1 hingga 60 detik, tergantung pada metrik. Metrik tidak akan dikembalikan jika `Period` bidang memiliki nilai kurang dari periode penerbitan untuk metrik.

6. Jalankan `aws cloudwatch get-metric-data` dengan file konfigurasi yang dibuat pada langkah sebelumnya. Contoh diberikan di bawah ini.

```
aws cloudwatch get-metric-data --region ground-station-region-code --cli-input-json
 file://<nameOfConfigurationFileCreatedInStep2>.json
```

Metrik akan diberikan stempel waktu dari kontak Anda. Contoh keluaran AWS Ground Station metrik disediakan di bawah ini.


```
{
   "MetricDataResults": [
      {
         "Id": "receivedPowerExample",
         "Label": "ReceivedPowerExample",
         "Timestamps": [
            "2024-04-08T18:35:00+00:00",
            "2024-04-08T18:30:00+00:00",
            "2024-04-08T18:25:00+00:00"
         ],
         "Values": [
            -33.30191555023193,
            -31.46100273132324,
            -32.13915576934814
         ],
         "StatusCode": "Complete"
      }
   ],
   "Messages": []
}
```

Keamanan di AWS Ground Station

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda akan mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan. AWS menyediakan alat dan fitur khusus keamanan untuk membantu Anda memenuhi tujuan keamanan Anda. Alat dan fitur ini termasuk keamanan jaringan, manajemen konfigurasi, kontrol akses, dan keamanan data.

Saat menggunakan AWS Ground Station, kami menyarankan Anda mengikuti praktik terbaik industri dan menerapkan end-to-end enkripsi. AWS menyediakan APIs bagi Anda untuk mengintegrasikan enkripsi dan perlindungan data. Untuk informasi selengkapnya tentang AWS keamanan, lihat whitepaper Pengantar AWS Security.

Gunakan topik berikut untuk mempelajari cara mengamankan sumber daya Anda.

Topik

Identity and Access Management untuk AWS Ground Station

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS Ground Station IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Contoh kebijakan berbasis identitas untuk AWS Ground Station
- Memecahkan masalah AWS Ground Station identitas dan akses

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda berdasarkan peran Anda:

- Pengguna layanan - minta izin dari administrator Anda jika Anda tidak dapat mengakses fitur (lihat Memecahkan masalah AWS Ground Station identitas dan akses)
- Administrator layanan - tentukan akses pengguna dan mengirimkan permintaan izin (lihat Bagaimana AWS Ground Station bekerja dengan IAM)
- Administrator IAM - tulis kebijakan untuk mengelola akses (lihat Contoh kebijakan berbasis identitas untuk AWS Ground Station)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredensional dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), autentikasi masuk tunggal, atau kredensional. Google/Facebook Untuk informasi selengkapnya tentang cara masuk, lihat Cara masuk ke Akun AWS Anda dalam Panduan Pengguna AWS Sign-In .

Untuk akses terprogram, AWS sediakan SDK dan CLI untuk menandatangani permintaan secara kriptografis. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensial pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas terfederasi

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredensi dari sumber identitas. Identitas terfederasi mengambil peran yang memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami menyarankan AWS IAM Identity Center. Untuk informasi selengkapnya, lihat Apa itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensial sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses AWS menggunakan kredensi sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan mempermudah pengelolaan izin untuk pengguna dalam jumlah besar. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensial sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI atau AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna terfederasi, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya.

AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan principal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diambil oleh pengguna. Kebijakan IAM mendefinisikan izin terlepas dari metode yang Anda gunakan untuk melakukan operasinya.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang bisa dilakukan oleh identitas tersebut, terhadap sumber daya yang mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan yang dikelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat berupa kebijakan inline (disematkan langsung ke dalam satu identitas) atau kebijakan terkelola (kebijakan mandiri yang dilampirkan pada banyak identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan kebijakan inline, lihat Pilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat menetapkan izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batasan izin – Menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — Tentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam Panduan Pengguna AWS Organizations .
- Kebijakan kontrol sumber daya (RCPs) — Tetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan lanjutan yang diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna terfederasi. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Bagaimana AWS Ground Station bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses AWS Ground Station, pelajari fitur IAM yang tersedia untuk digunakan. AWS Ground Station

Fitur IAM yang dapat Anda gunakan AWS Ground Station

Fitur IAM	AWS Ground Station dukungan
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya

Fitur IAM	AWS Ground Station dukungan
kunci-kunci persyaratan kebijakan (spesifik layanan)	Ya
ACLs	Tidak
ABAC (tanda dalam kebijakan)	Ya
Kredensial sementara	Ya
Izin principal	Ya
Peran layanan	Tidak
Peran terkait layanan	Ya

Untuk mendapatkan tampilan tingkat tinggi tentang cara AWS Ground Station dan AWS layanan lain bekerja dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk AWS Ground Station

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk AWS Ground Station

Untuk melihat contoh kebijakan AWS Ground Station berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Ground Station

Kebijakan berbasis sumber daya dalam AWS Ground Station

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh principal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan principal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai principal dalam kebijakan berbasis sumber daya. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk AWS Ground Station

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar AWS Ground Station tindakan, lihat Tindakan yang ditentukan oleh AWS Ground Station dalam Referensi Otorisasi Layanan.

Tindakan kebijakan AWS Ground Station menggunakan awalan berikut sebelum tindakan:

```
groundstation
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
      "groundstation:action1",
      "groundstation:action2"
         ]
```

Untuk melihat contoh kebijakan AWS Ground Station berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Ground Station

Sumber daya kebijakan untuk AWS Ground Station

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Untuk melihat daftar jenis sumber daya dan jenis AWS Ground Station sumber daya ARNs, lihat Sumber daya yang ditentukan oleh AWS Ground Station dalam Referensi Otorisasi Layanan. Untuk mempelajari dengan tindakan mana Anda dapat menentukan ARN setiap sumber daya, lihat Tindakan yang ditentukan oleh AWS Ground Station.

Untuk melihat contoh kebijakan AWS Ground Station berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Ground Station

Kunci kondisi kebijakan untuk AWS Ground Station

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Condition` menentukan ketika pernyataan dieksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci AWS Ground Station kondisi, lihat Kunci kondisi untuk AWS Ground Station dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang ditentukan oleh AWS Ground Station.

Untuk melihat contoh kebijakan AWS Ground Station berbasis identitas, lihat. Contoh kebijakan berbasis identitas untuk AWS Ground Station

ACLs di AWS Ground Station

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan AWS Ground Station

Mendukung ABAC (tanda dalam kebijakan): Ya

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut tanda. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi saat tag prinsipal cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredensi sementara dengan AWS Ground Station

Mendukung kredensial sementara: Ya

Kredensi sementara menyediakan akses jangka pendek ke AWS sumber daya dan secara otomatis dibuat saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM dalam Panduan Pengguna IAM.

Izin utama lintas layanan untuk AWS Ground Station

Mendukung sesi akses terusan (FAS): Ya

Sesi akses teruskan (FAS) menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses terusan.

Peran layanan untuk AWS Ground Station

Mendukung peran layanan: Tidak

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak AWS Ground Station fungsionalitas. Edit peran layanan hanya jika AWS Ground Station memberikan panduan untuk melakukannya.

Peran terkait layanan untuk AWS Ground Station

Mendukung peran terkait layanan: Ya

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Contoh kebijakan berbasis identitas untuk AWS Ground Station

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau mengubah sumber daya AWS Ground Station . Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh AWS Ground Station, termasuk format ARNs untuk setiap jenis sumber daya, lihat Kunci tindakan, sumber daya, dan kondisi untuk AWS Ground Station dalam Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- Menggunakan AWS Ground Station konsol
- Mengizinkan pengguna melihat izin mereka sendiri

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus AWS Ground Station sumber daya di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan

yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.

- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Menggunakan AWS Ground Station konsol

Untuk mengakses AWS Ground Station konsol, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang AWS Ground Station sumber

daya di Anda Akun AWS. Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan AWS Ground Station konsol, lampirkan juga kebijakan AWS Ground Station *ConsoleAccess* atau *ReadOnly* AWS terkelola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
```

```
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Memecahkan masalah AWS Ground Station identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS Ground Station dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di AWS Ground Station
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS Ground Station sumber daya saya

Saya tidak berwenang untuk melakukan tindakan di AWS Ground Station

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `groundstation:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 groundstation:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `groundstation:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS Ground Station.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di AWS Ground Station. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS Ground Station sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah AWS Ground Station mendukung fitur-fitur ini, lihatBagaimana AWS Ground Station bekerja dengan IAM.

- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

AWS kebijakan terkelola untuk AWS Ground Station

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: AWSGround StationAgentInstancePolicy

Anda dapat melampirkan kebijakan `AWSGroundStationAgentInstancePolicy` ke identitas IAM Anda.

Kebijakan ini memberikan izin AWS Ground Station Agen ke instans Amazon EC2 Anda yang memungkinkan instans mengirim dan menerima data selama kontak Ground Station. Semua izin dalam kebijakan ini berasal dari layanan Ground Station.

Detail izin

Kebijakan ini mencakup izin berikut.

- `groundstation`— Memungkinkan instance titik akhir aliran data untuk memanggil Agen Ground Station. APIs

Untuk melihat versi terbaru dokumen kebijakan JSON, lihat AWSGroundStationAgentInstancePolicydi Panduan Referensi Kebijakan Terkelola AWS.

AWS kebijakan terkelola: AWSService RoleForGroundStationDataflowEndpointGroupPolicy

Anda tidak dapat melampirkan AWSService RoleForGroundStationDataflowEndpointGroupPolicy ke entitas IAM Anda. Kebijakan ini dilampirkan pada peran terkait layanan yang memungkinkan AWS Ground Station untuk melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran terkait layanan.

Kebijakan ini memberikan izin EC2 yang memungkinkan AWS Ground Station untuk menemukan alamat publik. IPv4

Detail izin

Kebijakan ini mencakup izin berikut.

- `ec2:DescribeAddresses`— Memungkinkan AWS Ground Station untuk membuat daftar semua IPs yang terkait dengan EIPs atas nama Anda.

- `ec2:DescribeNetworkInterfaces`— Memungkinkan AWS Ground Station untuk mendapatkan informasi tentang antarmuka jaringan yang terkait dengan instans EC2 atas nama Anda.

Untuk melihat versi terbaru dokumen kebijakan JSON, lihat AWSServiceRoleForGroundStationDataflowEndpointGroupPolicydi Panduan Referensi Kebijakan Terkelola AWS.

AWS Ground Station pembaruan kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola AWS Ground Station sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman Riwayat AWS Ground Station dokumen.

Ubah	Deskripsi	Tanggal
AWSGroundStationAgentInstancePolicy — Permbaruan ke kebijakan yang sudah ada	AWS Ground Station menambahkan izin baru untuk memungkinkan agen mengambil respons URLs tugas untuk operasi Ground Station yang ditingkatkan.	November 13, 2025
AWSGroundStationAgentInstancePolicy – Kebijakan baru	AWS Ground Station menambahkan kebijakan baru untuk memberikan izin instans titik akhir aliran data untuk menggunakan AWS Ground Station Agent.	12 April 2023
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy – Kebijakan baru	AWS Ground Station menambahkan kebijakan baru yang memberikan izin EC2 untuk memungkinkan AWS Ground Station menemukan	November 02, 2022

Ubah	Deskripsi	Tanggal
	alamat IPv4 publik yang terkait dengan EIP dan antarmuka jaringan yang terkait dengan instans EC2.	
AWS Ground Station mulai melacak perubahan	AWS Ground Station mulai melacak perubahan untuk kebijakan AWS terkelola.	Maret 01, 2021

Gunakan peran terkait layanan untuk Ground Station

AWS Ground Station menggunakan AWS Identity and Access Management peran terkait layanan (IAM). Peran terkait layanan adalah jenis peran IAM unik yang terhubung langsung ke Ground Station. Peran terkait layanan telah ditentukan sebelumnya oleh Ground Station dan mencakup semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran terkait layanan membuat pengaturan Ground Station lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Ground Station mendefinisikan izin peran terkait layanan, dan kecuali ditentukan lain, hanya Ground Station yang dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, silakan lihat layanan AWS yang bisa digunakan dengan IAM dan carilah layanan yang memiliki opsi Ya di kolom Peran terkait layanan. Pilih Ya dengan sebuah tautan untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Izin peran terkait layanan untuk Ground Station

Ground Station menggunakan peran terkait layanan bernama — AWSServiceRoleForGroundStationDataflowEndpointGroupAWS GroundStation menggunakan peran terkait layanan ini untuk memanggil EC2 guna menemukan alamat publik. IPv4

Peran AWSService RoleForGroundStationDataflowEndpointGroup terkait layanan mempercayai layanan berikut untuk mengambil peran:

- `groundstation.amazonaws.com`

Kebijakan izin peran bernama AWSService RoleForGroundStationDataflowEndpointGroupPolicy memungkinkan Ground Station menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

- Tindakan: `ec2:DescribeAddresses` pada `all AWS resources (*)`

  Action memungkinkan Ground Station untuk daftar semua IPs yang terkait dengan EIPs.

- Tindakan: `ec2:DescribeNetworkInterfaces` pada `all AWS resources (*)`

  Tindakan memungkinkan Ground Station untuk mendapatkan informasi tentang antarmuka jaringan yang terkait dengan instans EC2

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

Membuat peran terkait layanan untuk Ground Station

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda membuat DataflowEndpointGroup di dalam AWS CLI atau AWS API, Ground Station membuat peran terkait layanan untuk Anda.

Jika Anda menghapus peran terkait layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda membuat DataflowEndpointGroup, Ground Station menciptakan peran terkait layanan untuk Anda lagi.

Anda juga dapat menggunakan konsol IAM untuk membuat peran terkait layanan dengan kasus penggunaan Pengiriman Data ke Amazon EC2. Di AWS CLI atau AWS API, buat peran terkait layanan dengan nama `groundstation.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 Ground Station

Ground Station tidak memungkinkan Anda untuk mengedit peran AWSService RoleForGroundStationDataflowEndpointGroup terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin merujuk peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk Ground Station

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, sebaiknya hapus peran tersebut. Dengan begitu, Anda tidak perlu lagi memantau atau memelihara entitas yang tidak digunakan.

Anda dapat menghapus peran terkait layanan hanya setelah pertama kali menghapus DataflowEndpointGroups menggunakan peran terkait layanan. Ini melindungi Anda dari pencabutan izin secara tidak sengaja ke Anda. DataflowEndpointGroups Jika peran terkait layanan digunakan dengan beberapa peran DataflowEndpointGroups, Anda harus menghapus semua DataflowEndpointGroups yang menggunakan peran terkait layanan sebelum dapat menghapusnya.

> **Note**
>
> Jika layanan Ground Station menggunakan peran saat Anda mencoba menghapus sumber daya, maka penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Untuk menghapus sumber daya Ground Station yang digunakan oleh AWSService RoleForGroundStationDataflowEndpointGroup

- Hapus DataflowEndpointGroups melalui AWS CLI atau AWS API.

Untuk menghapus peran tertaut layanan secara manual menggunakan IAM

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran AWSService RoleForGroundStationDataflowEndpointGroup terkait layanan. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan dalam Panduan Pengguna IAM.

Wilayah yang didukung untuk peran terkait layanan Ground Station

Ground Station mendukung penggunaan peran terkait layanan di semua wilayah di mana layanan tersedia. Untuk informasi selengkapnya, lihat Tabel Wilayah.

Pemecahan masalah

`NOT_AUTHORIZED_TO_CREATE_SLR`- Ini menunjukkan peran dalam akun Anda yang sedang digunakan untuk memanggil CreateDataflowEndpointGroup API tidak memiliki

`iam:CreateServiceLinkedRole` izin. Administrator dengan `iam:CreateServiceLinkedRole` izin harus secara manual membuat Peran Tertaut Layanan untuk akun Anda.

Enkripsi data saat istirahat untuk AWS Ground Station

AWS Ground Station menyediakan enkripsi secara default untuk melindungi data sensitif Anda saat istirahat menggunakan kunci enkripsi yang AWS dimiliki.

- AWS kunci yang dimiliki - AWS Ground Station menggunakan kunci ini secara default untuk secara otomatis mengenkripsi data pribadi yang dapat diidentifikasi secara langsung dan ephemerides. Anda tidak dapat melihat, mengelola, atau menggunakan kunci AWS milik, atau mengaudit penggunaannya; Namun, tidak perlu mengambil tindakan apa pun atau mengubah program untuk melindungi kunci yang mengenkripsi data. Untuk informasi selengkapnya, lihat kunci AWS yang dimiliki di Panduan AWS Key Management Service Pengembang.

Enkripsi data saat istirahat secara default membantu dengan mengurangi overhead operasional dan kompleksitas yang terlibat dalam melindungi data sensitif. Pada saat yang sama, ini memungkinkan membangun aplikasi aman yang memenuhi kepatuhan enkripsi yang ketat, serta persyaratan peraturan.

AWS Ground Station memberlakukan enkripsi pada semua data sensitif, saat istirahat, namun, untuk beberapa AWS Ground Station sumber daya, seperti ephemerides, Anda dapat memilih untuk menggunakan kunci yang dikelola pelanggan sebagai pengganti kunci terkelola default. AWS

- Kunci terkelola pelanggan - AWS Ground Station mendukung penggunaan kunci terkelola pelanggan simetris yang Anda buat, miliki, dan kelola menggantikan enkripsi yang AWS dimiliki yang ada. Karena Anda memiliki kontrol penuh atas lapisan enkripsi ini, Anda dapat melakukan tugas-tugas seperti:
  - Menetapkan dan memelihara kebijakan utama
  - Menetapkan dan memelihara kebijakan dan hibah IAM
  - Mengaktifkan dan menonaktifkan kebijakan utama
  - Memutar bahan kriptografi kunci
  - Menambahkan tanda
  - Membuat alias kunci
  - Kunci penjadwalan untuk penghapusan

Untuk informasi selengkapnya, lihat kunci terkelola pelanggan di Panduan AWS Key Management Service Pengembang.

Tabel berikut merangkum sumber daya yang AWS Ground Station mendukung penggunaan Customer Managed Keys

Jenis data	AWS enkripsi kunci yang dimiliki	Enkripsi kunci yang dikelola pelanggan (Opsional)
Data Ephemeris digunakan untuk menghitung lintasan Satelit	Diaktifkan	Diaktifkan
Ephemeris elevasi Azimuth digunakan untuk memerinta hkan antena	Diaktifkan	Diaktifkan

> **Note**
>
> AWS Ground Station secara otomatis mengaktifkan enkripsi saat istirahat menggunakan Kunci milik AWS untuk melindungi data yang dapat diidentifikasi secara pribadi tanpa biaya. Namun, AWS KMS biaya berlaku untuk menggunakan kunci yang dikelola pelanggan. Untuk informasi selengkapnya tentang harga, lihat AWS Key Management Service harga.
> Untuk informasi selengkapnya AWS KMS, lihat Panduan AWS Key Management Service Pengembang.

Untuk informasi spesifik untuk setiap jenis sumber daya, lihat:

- Enkripsi saat istirahat untuk data ephemeris TLE dan OEM
- Enkripsi saat istirahat untuk ephemeris elevasi azimuth

Buat kunci terkelola pelanggan

Anda dapat membuat kunci terkelola pelanggan simetris dengan menggunakan Konsol Manajemen AWS, atau. AWS KMS APIs

Untuk membuat kunci terkelola pelanggan simetris

Ikuti langkah-langkah untuk membuat kunci terkelola pelanggan simetris di Panduan AWS Key Management Service Pengembang.

Ikhtisar kebijakan utama

Kebijakan utama mengontrol akses ke kunci yang dikelola pelanggan Anda. Setiap kunci yang dikelola pelanggan harus memiliki persis satu kebijakan utama, yang berisi pernyataan yang menentukan siapa yang dapat menggunakan kunci dan bagaimana mereka dapat menggunakannya. Saat membuat kunci terkelola pelanggan, Anda dapat menentukan kebijakan kunci. Untuk informasi selengkapnya, lihat Mengelola akses ke kunci terkelola pelanggan di Panduan AWS Key Management Service Pengembang.

Untuk menggunakan kunci terkelola pelanggan dengan AWS Ground Station sumber daya, Anda harus mengonfigurasi kebijakan kunci untuk memberikan izin yang sesuai ke AWS Ground Station layanan. Izin spesifik dan konfigurasi kebijakan bergantung pada jenis sumber daya yang Anda enkripsi:

- Untuk data ephemeris TLE dan OEM - Lihat persyaratan dan Enkripsi saat istirahat untuk data ephemeris TLE dan OEM contoh kebijakan utama yang spesifik.
- Untuk data ephemeris elevasi azimuth - Lihat Enkripsi saat istirahat untuk ephemeris elevasi azimuth untuk persyaratan dan contoh kebijakan utama yang spesifik.

> **Note**
>
> Konfigurasi kebijakan kunci berbeda antara tipe ephemeris. Data ephemeris TLE dan OEM menggunakan hibah untuk akses kunci, sedangkan ephemeris elevasi azimuth menggunakan izin kebijakan kunci langsung. Pastikan Anda mengonfigurasi kebijakan kunci sesuai dengan jenis sumber daya tertentu yang Anda enkripsi.

Untuk informasi selengkapnya tentang menentukan izin dalam kebijakan dan akses kunci pemecahan masalah, lihat Panduan Pengembang. AWS Key Management Service

Menentukan kunci yang dikelola pelanggan untuk AWS Ground Station

Anda dapat menentukan kunci yang dikelola pelanggan untuk mengenkripsi sumber daya berikut:

- Ephemeris (TLE, OEM, dan elevasi azimuth)

Saat Anda membuat sumber daya, Anda dapat menentukan kunci data dengan menyediakan kmsKeyArn

- kmsKeyArn- Pengidentifikasi kunci untuk kunci yang dikelola AWS KMS pelanggan

AWS Ground Station konteks enkripsi

Konteks enkripsi adalah kumpulan opsional pasangan kunci-nilai yang berisi informasi kontekstual tambahan tentang data. AWS KMS menggunakan konteks enkripsi sebagai data otentikasi tambahan untuk mendukung enkripsi yang diautentikasi. Bila Anda menyertakan konteks enkripsi dalam permintaan untuk mengenkripsi data, AWS KMS mengikat konteks enkripsi ke data terenkripsi. Untuk mendekripsi data, Anda menyertakan konteks enkripsi yang sama dalam permintaan.

AWS Ground Station menggunakan konteks enkripsi yang berbeda tergantung pada sumber daya yang dienkripsi dan menentukan konteks enkripsi khusus untuk setiap hibah kunci yang dibuat.

Untuk detail konteks enkripsi khusus sumber daya, lihat:

- Enkripsi saat istirahat untuk data ephemeris TLE dan OEM
- Enkripsi saat istirahat untuk ephemeris elevasi azimuth

Enkripsi saat istirahat untuk data ephemeris TLE dan OEM

Persyaratan kebijakan utama untuk ephemeris TLE dan OEM

Untuk menggunakan kunci yang dikelola pelanggan dengan data ephemeris, kebijakan kunci Anda harus memberikan izin berikut ke layanan: AWS Ground Station

- `kms:CreateGrant`- Membuat hibah akses pada kunci yang dikelola pelanggan. Memberikan AWS Ground Station akses untuk melakukan operasi hibah pada kunci yang dikelola pelanggan untuk membaca dan menyimpan data terenkripsi.

- `kms:DescribeKey`- Memberikan rincian kunci yang dikelola pelanggan AWS Ground Station untuk memungkinkan memvalidasi kunci sebelum mencoba menggunakan kunci yang disediakan.

Untuk informasi selengkapnya tentang Menggunakan Hibah, lihat Panduan AWS Key Management Service Pengembang.

Izin pengguna IAM untuk membuat ephemeris dengan kunci yang dikelola pelanggan

Saat AWS Ground Station menggunakan kunci yang dikelola pelanggan dalam operasi kriptografi, ia bertindak atas nama pengguna yang membuat sumber daya ephemeris.

Untuk membuat sumber daya ephemeris menggunakan kunci yang dikelola pelanggan, pengguna harus memiliki izin untuk memanggil operasi berikut pada kunci yang dikelola pelanggan:

- `kms:CreateGrant`- Memungkinkan pengguna untuk membuat hibah pada kunci yang dikelola pelanggan atas nama. AWS Ground Station
- `kms:DescribeKey`- Memungkinkan pengguna untuk melihat detail kunci yang dikelola pelanggan untuk memvalidasi kunci.

Anda dapat menentukan izin yang diperlukan ini dalam kebijakan kunci, atau dalam kebijakan IAM jika kebijakan kunci memungkinkan hal tersebut. Izin ini memastikan bahwa pengguna dapat mengotorisasi AWS Ground Station untuk menggunakan kunci yang dikelola pelanggan untuk operasi enkripsi atas nama mereka.

Bagaimana AWS Ground Station menggunakan hibah AWS KMS untuk ephemeris

AWS Ground Station memerlukan hibah kunci untuk menggunakan kunci yang dikelola pelanggan Anda.

Saat Anda mengunggah ephemeris yang dienkripsi dengan kunci yang dikelola pelanggan, AWS Ground Station buat hibah kunci atas nama Anda dengan mengirimkan permintaan ke. CreateGrant AWS KMS Hibah AWS KMS digunakan untuk memberikan AWS Ground Station akses ke AWS KMS kunci di akun Anda.

Hal ini memungkinkan AWS Ground Station untuk melakukan hal berikut:

- Panggilan GenerateDataKeyuntuk menghasilkan kunci data terenkripsi dan menyimpannya, karena kunci data tidak segera digunakan untuk mengenkripsi.

- Panggil Dekripsi untuk menggunakan kunci data terenkripsi yang disimpan untuk mengakses data terenkripsi.
- Panggil Enkripsi untuk menggunakan kunci data untuk mengenkripsi data.
- Siapkan kepala sekolah yang pensiun untuk memungkinkan layanan. RetireGrant

Anda dapat mencabut akses ke hibah kapan saja. Jika Anda melakukannya, AWS Ground Station tidak akan dapat mengakses data apa pun yang dienkripsi oleh kunci yang dikelola pelanggan, yang memengaruhi operasi yang bergantung pada data tersebut. Misalnya, jika Anda menghapus hibah kunci dari ephemeris yang saat ini digunakan untuk kontak maka tidak AWS Ground Station akan dapat menggunakan data ephemeris yang disediakan untuk mengarahkan antena selama kontak. Ini akan menyebabkan kontak berakhir dalam keadaan GAGAL.

Konteks enkripsi Ephemeris

Hibah kunci untuk mengenkripsi sumber daya ephemeris terikat pada ARN satelit tertentu.

```
"encryptionContext": {
    "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
    "aws:s3:arn":
 "arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
}
```

> **Note**
>
> Hibah kunci digunakan kembali untuk pasangan kunci-satelit yang sama.

Menggunakan konteks enkripsi untuk pemantauan

Saat Anda menggunakan kunci terkelola pelanggan simetris untuk mengenkripsi ephemerides Anda, Anda juga dapat menggunakan konteks enkripsi dalam catatan audit dan log untuk mengidentifikasi bagaimana kunci yang dikelola pelanggan digunakan. Konteks enkripsi juga muncul di log yang dihasilkan oleh AWS CloudTrail atau Amazon CloudWatch Logs.

Menggunakan konteks enkripsi untuk mengontrol akses ke kunci terkelola pelanggan Anda

Anda dapat menggunakan konteks enkripsi dalam kebijakan utama dan kebijakan IAM `conditions` untuk mengontrol akses ke kunci terkelola pelanggan simetris Anda. Anda juga dapat menggunakan kendala konteks enkripsi dalam hibah.

AWS Ground Station menggunakan batasan konteks enkripsi dalam hibah untuk mengontrol akses ke kunci yang dikelola pelanggan di akun atau wilayah Anda. Batasan hibah mengharuskan operasi yang diizinkan oleh hibah menggunakan konteks enkripsi yang ditentukan.

Berikut ini adalah contoh pernyataan kebijakan kunci untuk memberikan akses ke kunci yang dikelola pelanggan untuk konteks enkripsi tertentu. Kondisi dalam pernyataan kebijakan ini mengharuskan hibah memiliki batasan konteks enkripsi yang menentukan konteks enkripsi.

Contoh berikut menunjukkan kebijakan kunci untuk data ephemeris yang terikat ke satelit:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow AWS Ground Station to Describe key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        },
        {
            "Sid": "Allow AWS Ground Station to Create Grant on key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
```

```
                    "kms:EncryptionContext:aws:groundstation:arn":
 "arn:aws:groundstation::123456789012:satellite/satellite-id"
                }
            }
        }
    ]
}
```

Memantau kunci enkripsi Anda untuk ephemeris

Saat Anda menggunakan kunci yang dikelola AWS Key Management Service pelanggan dengan sumber daya ephemeris, Anda dapat menggunakan atau CloudWatch log AWS CloudTrailAmazon untuk melacak permintaan yang AWS Ground Station dikirim. AWS KMS Contoh berikut adalah CloudTrail peristiwa untuk CreateGrant,, Dekripsi GenerateDataKey, dan DescribeKeyuntuk memantau AWS KMS operasi yang dipanggil oleh AWS Ground Station untuk mengakses data yang dienkripsi oleh kunci yang dikelola pelanggan Anda.

CreateGrant

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris Anda, AWS Ground Station kirimkan CreateGrantpermintaan atas nama Anda untuk mengakses AWS KMS kunci di akun Anda. AWS Hibah AWS Ground Station yang dibuat khusus untuk sumber daya yang terkait dengan kunci yang dikelola AWS KMS pelanggan. Selain itu, AWS Ground Station gunakan RetireGrantoperasi untuk menghapus hibah saat Anda menghapus sumber daya.

Contoh peristiwa berikut mencatat CreateGrantoperasi untuk ephemeris:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "ASIAIOSFODNN7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
```

```
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-02-22T22:22:22Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2022-02-22T22:22:22Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "operations": [
            "GenerateDataKeyWithoutPlaintext",
            "Decrypt",
            "Encrypt"
        ],
        "constraints": {
            "encryptionContextSubset": {
                "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
            }
        },
        "granteePrincipal": "groundstation.us-west-2.amazonaws.com",
        "retiringPrincipal": "groundstation.us-west-2.amazonaws.com",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
```

```
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

DescribeKey

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris Anda, AWS Ground Station kirimkan DescribeKeypermintaan atas nama Anda untuk memvalidasi bahwa kunci yang diminta ada di akun Anda.

Contoh peristiwa berikut mencatat DescribeKeyoperasi:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/User/Role",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "ASIAIOSFODNN7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Role",
                "accountId": "111122223333",
                "userName": "User"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-02-22T22:22:22Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "AWS Internal"
    },
```

```
    "eventTime": "2022-02-22T22:22:22Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

GenerateDataKey

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris Anda, AWS Ground Station kirimkan GenerateDataKeypermintaan untuk menghasilkan kunci data yang dapat digunakan untuk mengenkripsi data Anda.

Contoh peristiwa berikut mencatat GenerateDataKeyoperasi untuk ephemeris:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2022-02-22T22:22:22Z",
```

```
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keySpec": "AES_256",
        "encryptionContext": {
            "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
            "aws:s3:arn":
 "arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
        },
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventCategory": "Management"
}
```


Decrypt

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris Anda, AWS Ground Station gunakan operasi Dekripsi untuk mendekripsi ephemeris yang disediakan jika sudah dienkripsi dengan kunci terkelola pelanggan yang sama. Misalnya jika ephemeris sedang diunggah dari bucket S3 dan dienkripsi dalam ember itu dengan kunci yang diberikan.

Contoh peristiwa berikut mencatat operasi Dekripsi untuk ephemeris:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2022-02-22T22:22:22Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "encryptionContext": {
            "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
            "aws:s3:arn":
 "arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
        },
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventCategory": "Management"
}
```

Enkripsi saat istirahat untuk ephemeris elevasi azimuth

Persyaratan kebijakan utama untuk ephemeris elevasi azimuth

Untuk menggunakan kunci terkelola pelanggan dengan data ephemeris elevasi azimuth, kebijakan kunci Anda harus memberikan izin berikut ke layanan. AWS Ground Station Tidak seperti data ephemeris TLE dan OEM yang menggunakan hibah, ephemeris elevasi azimuth menggunakan izin kebijakan kunci langsung untuk operasi enkripsi. Ini adalah metode yang lebih sederhana untuk mengelola izin, dan menggunakan kunci Anda.

- `kms:GenerateDataKey`- Menghasilkan kunci data untuk mengenkripsi data ephemeris elevasi azimuth Anda.
- `kms:Decrypt`- Mendekripsi kunci data terenkripsi saat mengakses data ephemeris elevasi azimuth Anda.

Contoh kebijakan kunci yang memberikan AWS Ground Station akses ke kunci yang dikelola pelanggan

> **Note**
>
> Dengan azimuth elevation ephemeris, Anda harus mengonfigurasi izin ini secara langsung di kebijakan utama. Prinsipal AWS Ground Station layanan regional (misalnya,`groundstation.`*`region`*`.amazonaws.com`) harus diberikan izin ini dalam pernyataan kebijakan utama Anda. Tanpa pernyataan ini ditambahkan ke kebijakan utama tidak AWS Ground Station akan dapat menyimpan atau mengakses ephemeris elevasi azimuth kustom Anda.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow AWS Ground Station to Describe key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
```

```
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        },
        {
            "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*"
        }
    ]
}
```

Izin pengguna IAM untuk membuat ephemeris elevasi azimuth dengan kunci yang dikelola pelanggan

Saat AWS Ground Station menggunakan kunci yang dikelola pelanggan dalam operasi kriptografi, ia bertindak atas nama pengguna yang membuat sumber daya ephemeris elevasi azimuth.

Untuk membuat sumber daya ephemeris elevasi azimuth menggunakan kunci terkelola pelanggan, pengguna harus memiliki izin untuk memanggil operasi berikut pada kunci yang dikelola pelanggan:

- `kms:GenerateDataKey`- Memungkinkan pengguna untuk menghasilkan kunci data untuk mengenkripsi data ephemeris elevasi azimuth.
- `kms:Decrypt`- Memungkinkan pengguna untuk mendekripsi kunci data saat mengakses data ephemeris elevasi azimuth.
- `kms:DescribeKey`- Memungkinkan pengguna untuk melihat detail kunci yang dikelola pelanggan untuk memvalidasi kunci.

Anda dapat menentukan izin yang diperlukan ini dalam kebijakan kunci, atau dalam kebijakan IAM jika kebijakan kunci memungkinkan hal tersebut. Izin ini memastikan bahwa pengguna dapat mengotorisasi AWS Ground Station untuk menggunakan kunci yang dikelola pelanggan untuk operasi enkripsi atas nama mereka.

Cara AWS Ground Station menggunakan kebijakan utama untuk ephemeris elevasi azimuth

Saat Anda memberikan data ephemeris elevasi azimuth dengan kunci yang dikelola pelanggan, AWS Ground Station gunakan kebijakan utama untuk mengakses kunci enkripsi Anda. Izin diberikan langsung melalui pernyataan kebijakan utama daripada AWS Ground Station melalui hibah seperti dengan data ephemeris TLE atau OEM.

Jika Anda menghapus AWS Ground Station akses ke kunci yang dikelola pelanggan, AWS Ground Station tidak akan dapat mengakses data apa pun yang dienkripsi oleh kunci tersebut, yang memengaruhi operasi yang bergantung pada data tersebut. Misalnya, jika Anda menghapus izin kebijakan kunci untuk ephemeris elevasi azimuth yang saat ini digunakan untuk kontak, tidak AWS Ground Station akan dapat menggunakan data elevasi azimuth yang disediakan untuk memerintahkan antena selama kontak. Ini akan menyebabkan kontak berakhir dalam keadaan GAGAL.

Konteks enkripsi ephemeris elevasi Azimuth

Saat AWS Ground Station menggunakan AWS KMS kunci Anda untuk mengenkripsi data ephemeris elevasi azimuth, layanan menentukan konteks enkripsi. Konteks enkripsi adalah data otentikasi tambahan (AAD) yang AWS KMS digunakan untuk memastikan integritas data. Ketika konteks enkripsi ditentukan untuk operasi enkripsi, layanan harus menentukan konteks enkripsi yang sama untuk operasi dekripsi. Jika tidak, dekripsi akan gagal. Konteks enkripsi juga ditulis ke CloudTrail log Anda untuk membantu Anda memahami mengapa AWS KMS kunci yang diberikan digunakan. CloudTrail Log Anda mungkin berisi banyak entri yang menjelaskan penggunaan AWS KMS kunci, tetapi konteks enkripsi di setiap entri log dapat membantu Anda menentukan alasan penggunaan tertentu tersebut.

AWS Ground Station menentukan konteks enkripsi berikut saat melakukan operasi kriptografi dengan kunci yang dikelola pelanggan Anda pada ephemeris elevasi azimuth:

```
{
    "encryptionContext": {
        "aws:groundstation:ground-station-id": "Ohio 1",
        "aws:groundstation:arn": "arn:aws:groundstation:us-
east-2:111122223333:ephemeris/00a770b0-082d-45a4-80ed-SAMPLE",
        "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/00a770b0-082d-45a4-80ed-
SAMPLE/raw"
    }
```

```
}
```

Konteks enkripsi berisi:

`aws:groundstation:ground-station-id`

Nama stasiun bumi yang terkait dengan ephemeris elevasi azimuth.

aws:groundstation: arn

ARN dari sumber daya ephemeris.

aws:s3: arn

ARN dari ephemeris disimpan di Amazon S3.

Menggunakan konteks enkripsi untuk mengontrol akses ke kunci terkelola pelanggan Anda

Anda dapat menggunakan pernyataan kondisi IAM untuk mengontrol AWS Ground Station akses ke kunci yang dikelola pelanggan Anda. Menambahkan pernyataan kondisi pada `kms:GenerateDataKey` dan `kms:Decrypt` tindakan membatasi stasiun bumi mana yang AWS KMS dapat digunakan.

Berikut ini adalah contoh pernyataan kebijakan utama untuk memberikan AWS Ground Station akses ke kunci yang dikelola pelanggan Anda di wilayah tertentu untuk stasiun bumi tertentu. Kondisi dalam pernyataan kebijakan ini mengharuskan semua mengenkripsi dan mendekripsi akses ke kunci yang menentukan konteks enkripsi yang cocok dengan kondisi dalam kebijakan kunci.

Contoh kebijakan kunci yang memberikan AWS Ground Station akses ke kunci yang dikelola pelanggan untuk stasiun bumi tertentu

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow AWS Ground Station to Describe key",
            "Effect": "Allow",
            "Principal": {
```

```
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        },
        {
            "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:aws:groundstation:ground-station-id":
 "specific-ground-station-name"
                }
            }
        }
    ]
}
```

Contoh kebijakan kunci yang memberikan AWS Ground Station akses ke kunci yang dikelola pelanggan untuk beberapa stasiun bumi

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow AWS Ground Station to Describe key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
```

```
            "Resource": "*"
        },
        {
            "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
            "Effect": "Allow",
            "Principal": {
                "Service": "groundstation.us-east-1.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "kms:EncryptionContext:aws:groundstation:ground-station-id":
 [
                        "specific-ground-station-name-1",
                        "specific-ground-station-name-2"
                    ]
                }
            }
        }
    ]
}
```

Memantau kunci enkripsi Anda untuk ephemeris elevasi azimuth

Saat Anda menggunakan kunci terkelola AWS KMS pelanggan dengan sumber daya ephemeris elevasi azimuth, Anda dapat menggunakan CloudTrailatau CloudWatch mencatat untuk melacak permintaan yang dikirim ke. AWS Ground Station AWS KMS Contoh berikut adalah CloudTrail peristiwa untuk GenerateDataKeydan Dekripsi untuk memantau AWS KMS operasi yang dipanggil oleh AWS Ground Station untuk mengakses data yang dienkripsi oleh kunci yang dikelola pelanggan Anda.

GenerateDataKey

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris elevasi azimuth Anda, AWS Ground Station kirimkan GenerateDataKeypermintaan ke untuk AWS KMS menghasilkan kunci data yang dapat digunakan untuk mengenkripsi data Anda.

Contoh peristiwa berikut mencatat GenerateDataKeyoperasi untuk ephemeris elevasi azimuth:


```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "ASIAIOSFODNN7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "attributes": {
                "creationDate": "2025-08-25T14:45:48Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2025-08-25T14:52:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keySpec": "AES_256",
        "encryptionContext": {
            "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
            "aws:groundstation:ground-station-id": "Ohio 1",
            "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
        },
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
```

```
    "requestID": "ef6f9a8f-8ef6-46a1-bdcb-123456SAMPLE",
    "eventID": "952842d4-1389-3232-b885-123456SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "8424f6b6-2280-4d1d-b9fd-0348b1546cba",
    "eventCategory": "Management"
}
```

Decrypt

Saat Anda menggunakan kunci yang dikelola AWS KMS pelanggan untuk mengenkripsi sumber daya ephemeris elevasi azimuth Anda, AWS Ground Station gunakan operasi Dekripsi untuk mendekripsi data ephemeris elevasi azimuth yang disediakan jika sudah dienkripsi dengan kunci terkelola pelanggan yang sama.

Contoh peristiwa berikut mencatat operasi Dekripsi untuk ephemeris elevasi azimuth:

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "ASIAIOSFODNN7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            }
```

```
        },
        "attributes": {
            "creationDate": "2025-08-25T14:45:48Z",
            "mfaAuthenticated": "false"
        }
    },
    "invokedBy": "AWS Internal",
    "eventTime": "2025-08-25T14:54:01Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "encryptionContext": {
            "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
            "aws:groundstation:ground-station-id": "Ohio 1",
            "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
        },
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "a2f46066-49fb-461a-93cb-123456SAMPLE",
    "eventID": "e997b426-e3ad-31c7-a308-123456SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "477b568e-7f56-4f04-905c-623ff146f30d",
    "eventCategory": "Management"
}
```

Enkripsi data selama transit untuk AWS Ground Station

AWS Ground Station menyediakan enkripsi secara default untuk melindungi data sensitif Anda selama transit. Data dapat dialirkan antara lokasi AWS Ground Station antena dan instans Amazon EC2 Anda dengan dua cara, tergantung pada konfigurasi profil misi.

- AWS Ground Station Agen
- Titik akhir aliran data

Setiap metode streaming data menangani enkripsi data dalam perjalanan secara berbeda. Bagian berikut menjelaskan setiap metode.

AWS Ground Station Aliran agen

AWS Ground Station Agen mengenkripsi alirannya menggunakan kunci yang dikelola pelanggan. AWS KMS AWS Ground Station Agen yang berjalan di instans Amazon EC2 Anda akan secara otomatis mendekripsi aliran untuk menyediakan data yang didekripsi.

AWS KMS Kunci yang digunakan untuk mengenkripsi aliran ditentukan saat membuat parameter `MissionProfile` dalam. `streamsKmsKey` Semua izin yang memberikan AWS Ground Station akses ke kunci ditangani melalui kebijakan AWS KMS kunci yang dilampirkan. `streamsKmsKey`

Aliran titik akhir aliran data

Aliran titik akhir Dataflow dienkripsi menggunakan Datagram Transport Layer Security (DTLS). Ini dilakukan dengan menggunakan sertifikat yang ditandatangani sendiri, dan tidak memerlukan konfigurasi tambahan.

Contoh konfigurasi profil misi

Contoh yang diberikan menunjukkan bagaimana mengambil satelit siaran publik dan membuat profil misi yang mendukungnya. Template yang dihasilkan disediakan untuk membantu Anda mengambil kontak satelit siaran publik dan untuk membantu Anda membuat keputusan tentang satelit Anda.

Topik

JPSS-1 - Public broadcast satellite (PBS) - Evaluasi

Bagian contoh ini cocok denganIkhtisar proses orientasi pelanggan. Ini memberikan analisis kompatibilitas singkat dengan AWS Ground Station dan menetapkan panggung untuk contoh-contoh spesifik yang mengikuti.

Seperti disebutkan di Satelit siaran publik bagian ini, Anda dapat menggunakan satelit tertentu, atau jalur komunikasi satelit, yang tersedia untuk umum. Pada bagian ini kami menjelaskan JPSS-1 dalam istilah. AWS Ground Station Sebagai referensi, kami menggunakan Joint Polar Satellite System 1 (JPSS-1) Spacecraft High Rate Data (HRD) ke Direct Broadcast Stations (DBS) Radio Frequency (RF) Interface Control Document (ICD) untuk melengkapi contoh. Juga, perlu dicatat bahwa JPSS-1 dikaitkan dengan NORAD ID 43013.

Satelit JPSS-1 menawarkan satu uplink dan tiga jalur komunikasi downlink langsung, seperti yang terlihat pada Gambar 1-1 dari ICD. Dari keempat jalur komunikasi ini, hanya jalur komunikasi downlink High Rate Data (HRD) tunggal yang tersedia untuk konsumsi publik. Berdasarkan ini, Anda akan melihat jalur ini akan memiliki data yang jauh lebih spesifik yang terkait dengannya juga. Keempat jalur tersebut adalah sebagai berikut:

- Jalur perintah (uplink) pada frekuensi MHz pusat 2067,27 dengan kecepatan data 2-128 kbps. Jalur ini tidak dapat diakses publik.
- Jalur telemetri (downlink) pada frekuensi MHz pusat 2247,5 dengan kecepatan data 1-524 kbps. Jalur ini tidak dapat diakses publik.

- Jalur SMD (downlink) pada frekuensi GHz tengah 26,7034 dengan kecepatan data 150-300 Mbps. Jalur ini tidak dapat diakses publik.
- RF untuk jalur HRD (downlink) pada frekuensi MHz pusat 7812 dengan kecepatan data 15 Mbps. Ini memiliki MHz bandwidth 30, dan adalah right-hand-circular-polarized. Saat Anda menggunakan JPSS-1 AWS Ground Station, ini adalah jalur komunikasi yang dapat Anda akses. Jalur komunikasi ini berisi data ilmu instrumen, data rekayasa instrumen, data telemetri instrumen, dan data rumah tangga pesawat ruang angkasa real-time.

Saat kami membandingkan jalur data potensial, kami melihat bahwa jalur perintah (uplink), telemetri (downlink), dan HRD (downlink) memenuhi frekuensi, bandwidth, dan kemampuan penggunaan bersamaan multi-saluran. AWS Ground Station Jalur SMD tidak kompatibel karena frekuensi pusat berada di luar jangkauan penerima yang ada. Untuk informasi selengkapnya tentang kemampuan yang didukung, lihatAWS Ground Station Kemampuan Situs.

> **Note**
>
> Karena jalur SMD tidak kompatibel AWS Ground Station dengannya tidak akan direpresentasikan dalam konfigurasi contoh.

> **Note**
>
> Karena jalur perintah (uplink) dan telemetri (downlink) tidak ditentukan dalam ICD, juga tidak tersedia untuk penggunaan umum, nilai yang diberikan saat digunakan adalah nosional.

Satelit siaran publik memanfaatkan pengiriman data Amazon S3

Contoh ini dibangun dari analisis yang dilakukan di JPSS-1 - Public broadcast satellite (PBS) - Evaluasi bagian panduan pengguna.

Untuk contoh ini, Anda harus mengasumsikan skenario -- Anda ingin menangkap jalur komunikasi HRD sebagai frekuensi menengah digital dan menyimpannya untuk pemrosesan batch masa depan. Ini menghemat sampel kuadratur fase (I/Q) frekuensi radio mentah (RF) setelah didigitalkan. Setelah data ada di bucket Amazon S3 Anda, Anda dapat mendemodulasi dan memecahkan kode data menggunakan perangkat lunak apa pun yang Anda inginkan. Lihat MathWorks Tutorial untuk contoh rinci pemrosesan. Setelah menggunakan contoh ini, Anda dapat mempertimbangkan untuk

menambahkan komponen harga EC2 spot Amazon untuk memproses data dan menurunkan biaya pemrosesan Anda secara keseluruhan.

Jalur komunikasi

Bagian ini Rencanakan jalur komunikasi aliran data Anda mewakili memulai.

Semua cuplikan template berikut termasuk dalam bagian Resources dari template. CloudFormation

```
Resources:
  # Resources that you would like to create should be placed within the Resources
 section.
```

> **Note**
>
> Untuk informasi selengkapnya tentang isi CloudFormation template, lihat bagian Template.

Mengingat skenario kami untuk mengirimkan jalur komunikasi tunggal ke Amazon S3, Anda tahu bahwa Anda akan memiliki satu jalur pengiriman asinkron. Per Pengiriman data asinkron bagian, Anda harus menentukan bucket Amazon S3.

```
  # The S3 bucket where AWS Ground Station will deliver the downlinked data.
  GroundStationS3DataDeliveryBucket:
    Type: AWS::S3::Bucket
    DeletionPolicy: Retain
    UpdateReplacePolicy: Retain
    Properties:
      # Results in a bucket name formatted like: aws-groundstation-data-{account id}-
{region}-{random 8 character string}
      BucketName: !Join ["-", ["aws-groundstation-data", !Ref AWS::AccountId, !Ref
 AWS::Region, !Select [0, !Split ["-", !Select [2, !Split ["/", !Ref AWS::StackId]]]]]]
```

Selain itu, Anda perlu membuat peran dan kebijakan yang sesuai AWS Ground Station untuk memungkinkan penggunaan bucket.

```
  # The IAM role that AWS Ground Station will assume to have permission find and write
  # data to your S3 bucket.
  GroundStationS3DataDeliveryRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Statement:
          - Action:
              - 'sts:AssumeRole'
            Effect: Allow
            Principal:
              Service:
                - groundstation.amazonaws.com
            Condition:
              StringEquals:
                "aws:SourceAccount": !Ref AWS::AccountId
              ArnLike:
                "aws:SourceArn": !Sub "arn:aws:groundstation:${AWS::Region}:
${AWS::AccountId}:config/s3-recording/*"

  # The S3 bucket policy that defines what actions AWS Ground Station can perform on
 your S3 bucket.
  GroundStationS3DataDeliveryBucketPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Statement:
          - Action:
              - 's3:GetBucketLocation'
            Effect: Allow
            Resource:
              - !GetAtt GroundStationS3DataDeliveryBucket.Arn
          - Action:
              - 's3:PutObject'
            Effect: Allow
            Resource:
              - !Join [ "/", [ !GetAtt GroundStationS3DataDeliveryBucket.Arn, "*" ] ]
      PolicyName: GroundStationS3DataDeliveryPolicy
      Roles:
        - !Ref GroundStationS3DataDeliveryRole
```

AWS Ground Station konfigurasi

Bagian ini Buat konfigurasi mewakili memulai.

Anda memerlukan konfigurasi pelacakan untuk mengatur preferensi Anda menggunakan autotrack. Memilih PREFERRED sebagai autotrack dapat meningkatkan kualitas sinyal, tetapi tidak diperlukan untuk memenuhi kualitas sinyal karena kualitas ephemeris JPSS-1 yang memadai.

```
 TrackingConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Tracking Config"
     ConfigData:
       TrackingConfig:
         Autotrack: "PREFERRED"
```

Berdasarkan jalur komunikasi, Anda harus menentukan konfigurasi antena-downlink untuk mewakili bagian satelit serta perekaman s3 untuk merujuk ke bucket Amazon S3 yang baru saja Anda buat.

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 JpssDownlinkDigIfAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Downlink DigIF Antenna Config"
     ConfigData:
       AntennaDownlinkConfig:
         SpectrumConfig:
           Bandwidth:
             Units: "MHz"
             Value: 30
           CenterFrequency:
             Units: "MHz"
             Value: 7812
           Polarization: "RIGHT_HAND"

 # The AWS Ground Station S3 Recording Config that defines the S3 bucket and IAM role
to use
```

```
  # when AWS Ground Station delivers the downlink data.
  S3RecordingConfig:
    Type: AWS::GroundStation::Config
    DependsOn: GroundStationS3DataDeliveryBucketPolicy
    Properties:
      Name: "JPSS S3 Recording Config"
      ConfigData:
        S3RecordingConfig:
          BucketArn: !GetAtt GroundStationS3DataDeliveryBucket.Arn
          RoleArn: !GetAtt GroundStationS3DataDeliveryRole.Arn
```

AWS Ground Station profil misi

Bagian ini Buat profil misi mewakili memulai.

Sekarang setelah Anda memiliki konfigurasi terkait, Anda dapat menggunakannya untuk membangun aliran data. Anda akan menggunakan default untuk parameter yang tersisa.

```
  # The AWS Ground Station Mission Profile that groups the above configurations to
define how to downlink data.
  JpssAsynchMissionProfile:
    Type: AWS::GroundStation::MissionProfile
    Properties:
      Name: "43013 JPSS Asynchronous Data"
      MinimumViableContactDurationSeconds: 180
      TrackingConfigArn: !Ref TrackingConfig
      DataflowEdges:
        - Source: !Ref JpssDownlinkDigIfAntennaConfig
          Destination: !Ref S3RecordingConfig
```

Menyatukannya

Dengan sumber daya di atas, Anda sekarang memiliki kemampuan untuk menjadwalkan kontak JPSS-1 untuk pengiriman data asinkron dari salah satu onboard Anda. AWS Ground Station AWS Ground Station Lokasi

Berikut ini adalah CloudFormation template lengkap yang mencakup semua sumber daya yang dijelaskan dalam bagian ini digabungkan menjadi satu template yang dapat langsung digunakan CloudFormation.

CloudFormation Template bernama `AquaSnppJpss-1TerraDigIfS3DataDelivery.yml` berisi bucket Amazon S3 dan AWS Ground Station sumber daya yang diperlukan untuk menjadwalkan kontak dan menerima data siaran langsung Sinyal/IP VITA-49.

Jika Aqua, SNPP, JPSS-1/NOAA-20, dan Terra tidak masuk ke akun Anda, lihat. Satelit onboard

> **Note**
>
> Anda dapat mengakses template dengan mengakses bucket Amazon S3 yang melakukan onboarding pelanggan menggunakan kredensi yang valid. AWS Tautan di bawah ini menggunakan bucket Amazon S3 regional. Ubah kode `us-west-2` wilayah untuk mewakili wilayah yang sesuai tempat Anda ingin membuat CloudFormation tumpukan.
> Selain itu, petunjuk berikut menggunakan YAMAL. Namun, template tersedia dalam format YAMAL dan JSON. Untuk menggunakan JSON, ganti ekstensi `.yml` file dengan `.json` saat mengunduh templat.

Untuk mengunduh templat menggunakan AWS CLI, gunakan perintah berikut:

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
AquaSnppJpss-1TerraDigIfS3DataDelivery.yml .
```

Anda dapat melihat dan mengunduh templat di konsol dengan menavigasi ke URL berikut di browser Anda:

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-
west-2/AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

Anda dapat menentukan template secara langsung CloudFormation menggunakan link berikut:

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

Satelit siaran publik menggunakan titik akhir aliran data (narrowband)

Contoh ini dibangun dari analisis yang dilakukan di JPSS-1 - Public broadcast satellite (PBS) - Evaluasi bagian panduan pengguna.

Untuk melengkapi contoh ini, Anda harus mengasumsikan skenario -- Anda ingin menangkap jalur komunikasi HRD sebagai frekuensi menengah digital (DiGIF) dan memprosesnya seperti yang diterima oleh aplikasi endpoint aliran data pada instans Amazon EC2 menggunakan SDR.

Jalur komunikasi

Bagian ini Rencanakan jalur komunikasi aliran data Anda mewakili memulai. Untuk contoh ini, Anda akan membuat dua bagian dalam CloudFormation template Anda: bagian Parameter dan Sumber Daya.

> **Note**
>
> Untuk informasi selengkapnya tentang isi CloudFormation template, lihat bagian Template.

Untuk bagian Parameter, Anda akan menambahkan parameter berikut. Anda akan menentukan nilai untuk ini saat membuat tumpukan melalui CloudFormation konsol.

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  ReceiverAMI:
    Description: The Ground Station DDX AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> Anda perlu membuat key pair, dan memberikan nama untuk parameter Amazon EC2EC2Key. Lihat Membuat key pair untuk instans Amazon EC2 Anda.

Selain itu, Anda harus memberikan ID AMI spesifik wilayah yang benar, saat membuat CloudFormation tumpukan. Lihat AWS Ground Station Gambar Mesin Amazon (AMIs).

Cuplikan template yang tersisa termasuk dalam bagian Resources dari template.CloudFormation

```
Resources:
  # Resources that you would like to create should be placed within the resource
 section.
```

Mengingat skenario kami untuk mengirimkan jalur komunikasi tunggal ke instans EC2, Anda akan memiliki satu jalur pengiriman sinkron. Per Pengiriman data sinkron bagian, Anda harus menyiapkan dan mengonfigurasi instans Amazon EC2 dengan aplikasi titik akhir aliran data, dan membuat satu atau beberapa grup titik akhir aliran data.

```
  # The EC2 instance that will send/receive data to/from your satellite using AWS
 Ground Station.
  ReceiverInstance:
    Type: AWS::EC2::Instance
    Properties:
      DisableApiTermination: false
      IamInstanceProfile: !Ref GeneralInstanceProfile
      ImageId: !Ref ReceiverAMI
      InstanceType: m5.4xlarge
      KeyName: !Ref EC2Key
      Monitoring: true
      PlacementGroupName: !Ref ClusterPlacementGroup
      SecurityGroupIds:
        - Ref: InstanceSecurityGroup
      SubnetId: !Ref ReceiverSubnet
      BlockDeviceMappings:
        - DeviceName: /dev/xvda
          Ebs:
            VolumeType: gp2
            VolumeSize: 40
      Tags:
        - Key: Name
          Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
      UserData:
        Fn::Base64:
```

```
          |
          #!/bin/bash
          exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
2>&1
          echo `date +'%F %R:%S'` "INFO: Logging Setup" >&2

          GROUND_STATION_DIR="/opt/aws/groundstation"
          GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
          STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"

          echo "Creating ${STREAM_CONFIG_PATH}"
          cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
          {
            "ddx_streams": [
              {
                "streamName": "Downlink",
                "maximumWanRate": 4000000000,
                "lanConfigDevice": "lo",
                "lanConfigPort": 50000,
                "wanConfigDevice": "eth1",
                "wanConfigPort": 55888,
                "isUplink": false
              }
            ]
          }
          STREAM_CONFIG

          echo "Waiting for dataflow endpoint application to start"
          while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

          echo "Configuring dataflow endpoint application streams"
          python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
          sleep 2
          python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

          exit 0

  # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
  # Station will use to send/receive data to/from your satellite.
  DataflowEndpointGroup:
    Type: AWS::GroundStation::DataflowEndpointGroup
    Properties:
```

```
      ContactPostPassDurationSeconds: 180
      ContactPrePassDurationSeconds: 120
      EndpointDetails:
        - Endpoint:
            Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
            Address:
              Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
              Port: 55888
          SecurityDetails:
            SecurityGroupIds:
              - Ref: "DataflowEndpointSecurityGroup"
            SubnetIds:
              - !Ref ReceiverSubnet
            RoleArn: !GetAtt DataDeliveryServiceRole.Arn

  # The security group for your EC2 instance.
  InstanceSecurityGroup:
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: AWS Ground Station receiver instance security group.
      VpcId: !Ref ReceiverVPC
      SecurityGroupIngress:
        # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
          Description: "AWS Ground Station Downlink Stream"

  # The security group that the ENI created by AWS Ground Station belongs to.
  DataflowEndpointSecurityGroup:
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
      VpcId: !Ref ReceiverVPC
      SecurityGroupEgress:
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 10.0.0.0/8
          Description: "AWS Ground Station Downlink Stream To 10/8"
```

```
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 172.16.0.0/12
          Description: "AWS Ground Station Downlink Stream To 172.16/12"
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 192.168.0.0/16
          Description: "AWS Ground Station Downlink Stream To 192.168/16"

  # The placement group in which your EC2 instance is placed.
  ClusterPlacementGroup:
    Type: AWS::EC2::PlacementGroup
    Properties:
      Strategy: cluster

  ReceiverVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: "10.0.0.0/16"
      Tags:
        - Key: "Name"
          Value: "AWS Ground Station - PBS to dataflow endpoint Example VPC"
        - Key: "Description"
          Value: "VPC for EC2 instance receiving AWS Ground Station data"

  ReceiverSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      # Ensure your CidrBlock will always have at least one available IP address per
dataflow endpoint.
      # See https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html for
subent sizing guidelines.
      CidrBlock: "10.0.0.0/24"
      Tags:
        - Key: "Name"
          Value: "AWS Ground Station - PBS to dataflow endpoint Example Subnet"
        - Key: "Description"
          Value: "Subnet for EC2 instance receiving AWS Ground Station data"
      VpcId: !Ref ReceiverVPC

  # An ENI providing a fixed IP address for AWS Ground Station to connect to.
  ReceiverInstanceNetworkInterface:
```

```
    Type: AWS::EC2::NetworkInterface
    Properties:
      Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
      GroupSet:
        - !Ref InstanceSecurityGroup
      SubnetId: !Ref ReceiverSubnet

  # Attach the ENI to the EC2 instance.
  ReceiverInstanceInterfaceAttachment:
    Type: AWS::EC2::NetworkInterfaceAttachment
    Properties:
      DeleteOnTermination: false
      DeviceIndex: "1"
      InstanceId: !Ref ReceiverInstance
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface
```

Selain itu, Anda juga perlu membuat kebijakan dan peran yang sesuai AWS Ground Station untuk memungkinkan Anda membuat elastic network interface (ENI) di akun Anda.

```
  # AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
  DataDeliveryServiceRole:
    Type: AWS::IAM::Role
    Properties:
      Policies:
        - PolicyDocument:
            Statement:
              - Action:
                  - ec2:CreateNetworkInterface
                  - ec2:DeleteNetworkInterface
                  - ec2:CreateNetworkInterfacePermission
                  - ec2:DeleteNetworkInterfacePermission
                  - ec2:DescribeSubnets
                  - ec2:DescribeVpcs
                  - ec2:DescribeSecurityGroups
                Effect: Allow
                Resource: '*'
            Version: '2012-10-17'
          PolicyName: DataDeliveryServicePolicy
      AssumeRolePolicyDocument:
```

```
      Version: 2012-10-17
      Statement:
        - Effect: Allow
          Principal:
            Service:
            - groundstation.amazonaws.com
          Action:
          - sts:AssumeRole

# The EC2 instance assumes this role.
InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Service:
              - "ec2.amazonaws.com"
          Action:
            - "sts:AssumeRole"
    Path: "/"
    ManagedPolicyArns:
      - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
      - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
      - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
      - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM

# The instance profile for your EC2 instance.
GeneralInstanceProfile:
  Type: AWS::IAM::InstanceProfile
  Properties:
    Roles:
      - !Ref InstanceRole
```

AWS Ground Station konfigurasi

Bagian ini Buat konfigurasi mewakili memulai.

Anda memerlukan konfigurasi pelacakan untuk mengatur preferensi Anda menggunakan autotrack. Memilih PREFERRED sebagai autotrack dapat meningkatkan kualitas sinyal, tetapi tidak diperlukan untuk memenuhi kualitas sinyal karena kualitas JPSS-1 ephemeris yang memadai.

```
 TrackingConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Tracking Config"
     ConfigData:
       TrackingConfig:
         Autotrack: "PREFERRED"
```

Berdasarkan jalur komunikasi, Anda harus menentukan konfigurasi antenna-downlink untuk mewakili bagian satelit, serta konfigurasi dataflow-endpoint untuk merujuk ke grup titik akhir aliran data yang mendefinisikan detail titik akhir.

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 SnppJpssDownlinkDigIfAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "SNPP JPSS Downlink DigIF Antenna Config"
     ConfigData:
       AntennaDownlinkConfig:
         SpectrumConfig:
           Bandwidth:
             Units: "MHz"
             Value: 30
           CenterFrequency:
             Units: "MHz"
             Value: 7812
           Polarization: "RIGHT_HAND"

 # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
 # from your satellite.
 DownlinkDigIfEndpointConfig:
   Type: AWS::GroundStation::Config
```

```
    Properties:
      Name: "Aqua SNPP JPSS Downlink DigIF Endpoint Config"
      ConfigData:
        DataflowEndpointConfig:
          DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
          DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station profil misi

Bagian ini Buat profil misi mewakili memulai.

Sekarang setelah Anda memiliki konfigurasi terkait, Anda dapat menggunakannya untuk membangun aliran data. Anda akan menggunakan default untuk parameter yang tersisa.

```
 # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
 # uplink and downlink data to your satellite.
 SnppJpssMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: "37849 SNPP And 43013 JPSS"
     ContactPrePassDurationSeconds: 120
     ContactPostPassDurationSeconds: 60
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Ref SnppJpssDownlinkDigIfAntennaConfig
         Destination: !Ref DownlinkDigIfEndpointConfig
```

Menyatukannya

Dengan sumber daya di atas, Anda sekarang memiliki kemampuan untuk menjadwalkan JPSS-1 kontak untuk pengiriman data sinkron dari salah satu AWS Ground StationAWS Ground Station Lokasi onboard Anda.

Berikut ini adalah CloudFormation template lengkap yang mencakup semua sumber daya yang dijelaskan dalam bagian ini digabungkan menjadi satu template yang dapat langsung digunakan CloudFormation.

CloudFormation Template bernama `AquaSnppJpssTerraDigIF.yml` dirancang untuk memberi Anda akses cepat untuk mulai menerima data frekuensi menengah digital (DiGIF) untuk satelit Aqua, SNPP, dan Terra. JPSS-1/NOAA-20 Ini berisi instans Amazon EC2 dan CloudFormation sumber daya yang diperlukan untuk menerima data siaran langsung DiGIF mentah.

Jika Aqua, SNPP, JPSS-1/NOAA-20, dan Terra tidak terhubung ke akun Anda, lihat. Satelit onboard

> **Note**
>
> Anda dapat mengakses template dengan mengakses bucket Amazon S3 yang melakukan onboarding pelanggan menggunakan kredensi yang valid.AWS Tautan di bawah ini menggunakan bucket Amazon S3 regional. Ubah kode `us-west-2` wilayah untuk mewakili wilayah yang sesuai tempat Anda ingin membuat CloudFormation tumpukan.
> Selain itu, petunjuk berikut menggunakan YAMAL. Namun, template tersedia dalam format YAMAL dan JSON. Untuk menggunakan JSON, ganti ekstensi `.yml` file dengan `.json` saat mengunduh templat.

Untuk mengunduh templat menggunakan AWS CLI, gunakan perintah berikut:

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
AquaSnppJpssTerraDigIF.yml .
```

Anda dapat melihat dan mengunduh templat di konsol dengan menavigasi ke URL berikut di browser Anda:

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-
west-2/AquaSnppJpssTerraDigIF.yml
```

Anda dapat menentukan template secara langsung CloudFormation menggunakan link berikut:

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
AquaSnppJpssTerraDigIF.yml
```

Sumber daya tambahan apa yang ditentukan oleh template?

`AquaSnppJpssTerraDigIF`Template mencakup sumber daya tambahan berikut:

- (Opsional) CloudWatch Event Triggers -AWS Lambda Fungsi yang dipicu menggunakan CloudWatch Peristiwa yang dikirim oleh AWS Ground Station sebelum dan sesudah kontak.AWS Lambda Fungsi akan memulai dan secara opsional menghentikan Instance Penerima Anda.
- (Opsional) Verifikasi EC2 untuk Kontak - Opsi untuk menggunakan Lambda untuk menyiapkan sistem verifikasi instans Amazon EC2 Anda untuk kontak dengan pemberitahuan SNS. Penting untuk dicatat bahwa ini mungkin dikenakan biaya tergantung pada penggunaan Anda saat ini.
- Ground Station Amazon Machine Image Retrieval Lambda - Opsi untuk memilih perangkat lunak apa yang diinstal dalam instans Anda dan AMI pilihan Anda. Opsi perangkat lunak termasuk aplikasi titik akhir aliran data, dengan atau tanpa radio yang ditentukan perangkat lunak. Opsi ini mungkin berubah saat pembaruan dan fitur perangkat lunak tambahan dirilis.
- Profil misi tambahan - Profil misi untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).
- Konfigurasi antena-downlink tambahan - Konfigurasi downlink antena untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).

Nilai dan parameter untuk satelit dalam template ini sudah terisi. Parameter ini memudahkan Anda untuk AWS Ground Station segera menggunakannya dengan satelit ini. Anda tidak perlu mengkonfigurasi nilai Anda sendiri untuk digunakan AWS Ground Station saat menggunakan template ini. Namun, Anda dapat menyesuaikan nilai untuk membuat template berfungsi untuk kasus penggunaan Anda.

Di mana saya menerima data saya?

Grup titik akhir aliran data diatur untuk menggunakan antarmuka jaringan instance penerima yang dibuat oleh bagian dari template. Instance penerima menggunakan aplikasi titik akhir aliran data untuk menerima aliran data dari AWS Ground Station port yang ditentukan oleh titik akhir aliran data. Setelah diterima, data tersedia untuk konsumsi melalui port UDP 50000 pada adaptor loopback dari instance penerima. Untuk informasi selengkapnya tentang menyiapkan grup titik akhir aliran data, lihat AWS::GroundStation::. DataflowEndpointGroup

Satelit siaran publik menggunakan titik akhir aliran data (didemodulasi dan diterjemahkan)

Contoh ini dibangun dari analisis yang dilakukan di JPSS-1 - Public broadcast satellite (PBS) - Evaluasi bagian panduan pengguna.

Untuk melengkapi contoh ini, Anda harus mengasumsikan skenario -- Anda ingin menangkap jalur komunikasi HRD sebagai data siaran langsung yang didemodulasi dan diterjemahkan menggunakan titik akhir aliran data. Contoh ini adalah titik awal yang baik jika Anda berencana untuk memproses data menggunakan perangkat lunak NASA Direct Readout Labs (RT-STPS dan IPOPP).

Jalur komunikasi

Bagian ini Rencanakan jalur komunikasi aliran data Anda mewakili memulai. Untuk contoh ini, Anda akan membuat dua bagian dalam CloudFormation template Anda: bagian Parameter dan Sumber Daya.

> **Note**
>
> Untuk informasi selengkapnya tentang isi CloudFormation template, lihat bagian Template.

Untuk bagian Parameter, Anda akan menambahkan parameter berikut. Anda akan menentukan nilai untuk ini saat membuat tumpukan melalui CloudFormation konsol.

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  ReceiverAMI:
    Description: The Ground Station DDX AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> Anda perlu membuat key pair, dan memberikan nama untuk parameter Amazon EC2EC2Key. Lihat Membuat key pair untuk instans Amazon EC2 Anda.

Selain itu, Anda harus memberikan ID AMI spesifik wilayah yang benar, saat membuat CloudFormation tumpukan. Lihat AWS Ground Station Gambar Mesin Amazon (AMIs).

Cuplikan template yang tersisa termasuk dalam bagian Resources dari template.CloudFormation

```
Resources:
  # Resources that you would like to create should be placed within the resource
 section.
```

Mengingat skenario kami untuk mengirimkan jalur komunikasi tunggal ke instans EC2, Anda akan memiliki satu jalur pengiriman sinkron. Per Pengiriman data sinkron bagian, Anda harus menyiapkan dan mengonfigurasi instans Amazon EC2 dengan aplikasi titik akhir aliran data, dan membuat satu atau beberapa grup titik akhir aliran data.

```
  # The EC2 instance that will send/receive data to/from your satellite using AWS
Ground Station.
  ReceiverInstance:
    Type: AWS::EC2::Instance
    Properties:
      DisableApiTermination: false
      IamInstanceProfile: !Ref GeneralInstanceProfile
      ImageId: !Ref ReceiverAMI
      InstanceType: m5.4xlarge
      KeyName: !Ref EC2Key
      Monitoring: true
      PlacementGroupName: !Ref ClusterPlacementGroup
      SecurityGroupIds:
        - Ref: InstanceSecurityGroup
      SubnetId: !Ref ReceiverSubnet
      BlockDeviceMappings:
        - DeviceName: /dev/xvda
          Ebs:
            VolumeType: gp2
            VolumeSize: 40
      Tags:
        - Key: Name
          Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
      UserData:
```

```
        Fn::Base64:
          |
          #!/bin/bash
          exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
2>&1
          echo `date +'%F %R:%S'` "INFO: Logging Setup" >&2

          GROUND_STATION_DIR="/opt/aws/groundstation"
          GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
          STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"

          echo "Creating ${STREAM_CONFIG_PATH}"
          cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
          {
            "ddx_streams": [
              {
                "streamName": "Downlink",
                "maximumWanRate": 4000000000,
                "lanConfigDevice": "lo",
                "lanConfigPort": 50000,
                "wanConfigDevice": "eth1",
                "wanConfigPort": 55888,
                "isUplink": false
              }
            ]
          }
          STREAM_CONFIG

          echo "Waiting for dataflow endpoint application to start"
          while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

          echo "Configuring dataflow endpoint application streams"
          python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
          sleep 2
          python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

          exit 0
```

```
 # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
 # Station will use to send/receive data to/from your satellite.
```

```
 DataflowEndpointGroup:
   Type: AWS::GroundStation::DataflowEndpointGroup
   Properties:
     ContactPostPassDurationSeconds: 180
     ContactPrePassDurationSeconds: 120
     EndpointDetails:
       - Endpoint:
           Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
           Address:
             Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
             Port: 55888
         SecurityDetails:
           SecurityGroupIds:
             - Ref: "DataflowEndpointSecurityGroup"
           SubnetIds:
             - !Ref ReceiverSubnet
           RoleArn: !GetAtt DataDeliveryServiceRole.Arn

 # The security group that the ENI created by AWS Ground Station belongs to.
 DataflowEndpointSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
     VpcId: !Ref ReceiverVPC
     SecurityGroupEgress:
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 10.0.0.0/8
         Description: "AWS Ground Station Downlink Stream To 10/8"
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 172.16.0.0/12
         Description: "AWS Ground Station Downlink Stream To 172.16/12"
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 192.168.0.0/16
         Description: "AWS Ground Station Downlink Stream To 192.168/16"

 # The placement group in which your EC2 instance is placed.
```

```
 ClusterPlacementGroup:
   Type: AWS::EC2::PlacementGroup
   Properties:
     Strategy: cluster

 # The security group for your EC2 instance.
 InstanceSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: AWS Ground Station receiver instance security group.
     VpcId: !Ref ReceiverVPC
     SecurityGroupIngress:
       # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
         Description: "AWS Ground Station Downlink Stream"

 ReceiverVPC:
   Type: AWS::EC2::VPC
   Properties:
     CidrBlock: "10.0.0.0/16"
     Tags:
       - Key: "Name"
         Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
VPC"
       - Key: "Description"
         Value: "VPC for EC2 instance receiving AWS Ground Station data"

 ReceiverSubnet:
   Type: AWS::EC2::Subnet
   Properties:
     CidrBlock: "10.0.0.0/24"
     Tags:
       - Key: "Name"
         Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
Subnet"
       - Key: "Description"
         Value: "Subnet for EC2 instance receiving AWS Ground Station data"
     VpcId: !Ref ReceiverVPC

 # An ENI providing a fixed IP address for AWS Ground Station to connect to.
```

```
 ReceiverInstanceNetworkInterface:
   Type: AWS::EC2::NetworkInterface
   Properties:
     Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
     GroupSet:
       - !Ref InstanceSecurityGroup
     SubnetId: !Ref ReceiverSubnet

 # Attach the ENI to the EC2 instance.
 ReceiverInstanceInterfaceAttachment:
   Type: AWS::EC2::NetworkInterfaceAttachment
   Properties:
     DeleteOnTermination: false
     DeviceIndex: "1"
     InstanceId: !Ref ReceiverInstance
     NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

 # The instance profile for your EC2 instance.
 GeneralInstanceProfile:
   Type: AWS::IAM::InstanceProfile
   Properties:
     Roles:
       - !Ref InstanceRole
```

Anda juga memerlukan kebijakan, peran, dan profil yang sesuai AWS Ground Station untuk memungkinkan Anda membuat elastic network interface (ENI) di akun Anda.

```
 # AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
 DataDeliveryServiceRole:
   Type: AWS::IAM::Role
   Properties:
     Policies:
       - PolicyDocument:
           Statement:
             - Action:
                 - ec2:CreateNetworkInterface
                 - ec2:DeleteNetworkInterface
                 - ec2:CreateNetworkInterfacePermission
                 - ec2:DeleteNetworkInterfacePermission
```

```
                - ec2:DescribeSubnets
                - ec2:DescribeVpcs
                - ec2:DescribeSecurityGroups
              Effect: Allow
              Resource: '*'
          Version: '2012-10-17'
        PolicyName: DataDeliveryServicePolicy
    AssumeRolePolicyDocument:
      Version: 2012-10-17
      Statement:
        - Effect: Allow
          Principal:
            Service:
            - groundstation.amazonaws.com
          Action:
          - sts:AssumeRole

# The EC2 instance assumes this role.
InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Service:
              - "ec2.amazonaws.com"
          Action:
            - "sts:AssumeRole"
    Path: "/"
    ManagedPolicyArns:
      - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
      - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
      - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
      - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM
```

AWS Ground Station konfigurasi

Bagian ini Buat konfigurasi mewakili panduan pengguna.

Anda memerlukan konfigurasi pelacakan untuk mengatur preferensi Anda menggunakan autotrack. Memilih PREFERRED sebagai autotrack dapat meningkatkan kualitas sinyal, tetapi tidak diperlukan untuk memenuhi kualitas sinyal karena kualitas JPSS-1 ephemeris yang memadai.

```
 TrackingConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Tracking Config"
     ConfigData:
       TrackingConfig:
         Autotrack: "PREFERRED"
```

Berdasarkan jalur komunikasi, Anda harus menentukan konfigurasi antenna-downlink-demod-decode untuk mewakili bagian satelit, serta konfigurasi dataflow-endpoint untuk merujuk ke grup titik akhir aliran data yang mendefinisikan detail titik akhir.

> **Note**
>
> Untuk detail tentang cara mengatur nilai untuk`DemodulationConfig`, dan`DecodeConfig`, silakan lihatAntena Downlink Demod Decode Config.

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 JpssDownlinkDemodDecodeAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Downlink Demod Decode Antenna Config"
     ConfigData:
       AntennaDownlinkDemodDecodeConfig:
         SpectrumConfig:
           CenterFrequency:
             Value: 7812
             Units: "MHz"
           Polarization: "RIGHT_HAND"
           Bandwidth:
             Value: 30
```

```
            Units: "MHz"
        DemodulationConfig:
          UnvalidatedJSON: '{
            "type":"QPSK",
            "qpsk":{
              "carrierFrequencyRecovery":{
                "centerFrequency":{
                  "value":7812,
                  "units":"MHz"
                },
                "range":{
                  "value":250,
                  "units":"kHz"
                }
              },
              "symbolTimingRecovery":{
                "symbolRate":{
                  "value":15,
                  "units":"Msps"
                },
                "range":{
                  "value":0.75,
                  "units":"ksps"
                },
                "matchedFilter":{
                  "type":"ROOT_RAISED_COSINE",
                  "rolloffFactor":0.5
                }
              }
            }
          }'
        DecodeConfig:
          UnvalidatedJSON: '{
            "edges":[
              {
                "from":"I-Ingress",
                "to":"IQ-Recombiner"
              },
              {
                "from":"Q-Ingress",
                "to":"IQ-Recombiner"
              },
              {
                "from":"IQ-Recombiner",
```

```
                "to":"CcsdsViterbiDecoder"
              },
              {
                "from":"CcsdsViterbiDecoder",
                "to":"NrzmDecoder"
              },
              {
                "from":"NrzmDecoder",
                "to":"UncodedFramesEgress"
              }
            ],
            "nodeConfigs":{
              "I-Ingress":{
                "type":"CODED_SYMBOLS_INGRESS",
                "codedSymbolsIngress":{
                  "source":"I"
                }
              },
              "Q-Ingress":{
                "type":"CODED_SYMBOLS_INGRESS",
                "codedSymbolsIngress":{
                  "source":"Q"
                }
              },
              "IQ-Recombiner":{
                "type":"IQ_RECOMBINER"
              },
              "CcsdsViterbiDecoder":{
                "type":"CCSDS_171_133_VITERBI_DECODER",
                "ccsds171133ViterbiDecoder":{
                  "codeRate":"ONE_HALF"
                }
              },
              "NrzmDecoder":{
                "type":"NRZ_M_DECODER"
              },
              "UncodedFramesEgress":{
                "type":"UNCODED_FRAMES_EGRESS"
              }
            }
          }'
```

```
 # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
 # from your satellite.
 DownlinkDemodDecodeEndpointConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "Aqua SNPP JPSS Downlink Demod Decode Endpoint Config"
     ConfigData:
       DataflowEndpointConfig:
         DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
         DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station profil misi

Bagian ini Buat profil misi mewakili panduan pengguna.

Sekarang setelah Anda memiliki konfigurasi terkait, Anda dapat menggunakannya untuk membangun aliran data. Anda akan menggunakan default untuk parameter yang tersisa.

```
 # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
 # uplink and downlink data to your satellite.
 SnppJpssMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: "37849 SNPP And 43013 JPSS"
     ContactPrePassDurationSeconds: 120
     ContactPostPassDurationSeconds: 60
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Join [ "/", [ !Ref JpssDownlinkDemodDecodeAntennaConfig,
"UncodedFramesEgress" ] ]
         Destination: !Ref DownlinkDemodDecodeEndpointConfig
```

Menyatukannya

Dengan sumber daya di atas, Anda sekarang memiliki kemampuan untuk menjadwalkan JPSS-1 kontak untuk pengiriman data sinkron dari salah satu AWS Ground StationAWS Ground Station Lokasi onboard Anda.

Berikut ini adalah CloudFormation template lengkap yang mencakup semua sumber daya yang dijelaskan dalam bagian ini digabungkan menjadi satu template yang dapat langsung digunakan CloudFormation.

CloudFormation Template bernama `AquaSnppJpss.yml` dirancang untuk memberi Anda akses cepat untuk mulai menerima data untuk Aqua, SNPP, dan JPSS-1/NOAA-20 satelit. Ini berisi instans Amazon EC2 dan AWS Ground Station sumber daya yang diperlukan untuk menjadwalkan kontak dan menerima data siaran langsung yang didemodulasi dan diterjemahkan.

Jika Aqua, SNPP, JPSS-1/NOAA-20, dan Terra tidak terhubung ke akun Anda, lihat. Satelit onboard

> **Note**
>
> Anda dapat mengakses template dengan mengakses bucket Amazon S3 yang melakukan onboarding pelanggan menggunakan kredensi yang valid.AWS Tautan di bawah ini menggunakan bucket Amazon S3 regional. Ubah kode `us-west-2` wilayah untuk mewakili wilayah yang sesuai tempat Anda ingin membuat CloudFormation tumpukan.
> Selain itu, petunjuk berikut menggunakan YAMAL. Namun, template tersedia dalam format YAMAL dan JSON. Untuk menggunakan JSON, ganti ekstensi `.yml` file dengan `.json` saat mengunduh templat.

Untuk mengunduh templat menggunakan AWS CLI, gunakan perintah berikut:

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml .
```

Anda dapat melihat dan mengunduh templat di konsol dengan menavigasi ke URL berikut di browser Anda:

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml
```

Anda dapat menentukan template secara langsung CloudFormation menggunakan link berikut:

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
AquaSnppJpss.yml
```

Sumber daya tambahan apa yang ditentukan oleh template?

`AquaSnppJpss`Template mencakup sumber daya tambahan berikut:

- (Opsional) CloudWatch Event Triggers -AWS Lambda Fungsi yang dipicu menggunakan CloudWatch Peristiwa yang dikirim oleh AWS Ground Station sebelum dan sesudah kontak.AWS Lambda Fungsi akan memulai dan secara opsional menghentikan Instance Penerima Anda.
- (Opsional) Verifikasi EC2 untuk Kontak - Opsi untuk menggunakan Lambda untuk menyiapkan sistem verifikasi instans Amazon EC2 Anda untuk kontak dengan pemberitahuan SNS. Penting untuk dicatat bahwa ini mungkin dikenakan biaya tergantung pada penggunaan Anda saat ini.
- Ground Station Amazon Machine Image Retrieval Lambda - Opsi untuk memilih perangkat lunak apa yang diinstal dalam instans Anda dan AMI pilihan Anda. Opsi perangkat lunak termasuk aplikasi titik akhir aliran data, dengan atau tanpa radio yang ditentukan perangkat lunak. Untuk menggunakan Pengiriman Data DiGIF Wideband dan Agen AWS Ground Station, lihat. Satelit siaran publik menggunakan AWS Ground Station Agen (pita lebar) Opsi ini mungkin berubah saat pembaruan dan fitur perangkat lunak tambahan dirilis.
- Profil misi tambahan - Profil misi untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).
- Konfigurasi antena-downlink tambahan - Konfigurasi downlink antena untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).

Nilai dan parameter untuk satelit dalam template ini sudah terisi. Parameter ini memudahkan Anda untuk AWS Ground Station segera menggunakannya dengan satelit ini. Anda tidak perlu mengkonfigurasi nilai Anda sendiri untuk digunakan AWS Ground Station saat menggunakan template ini. Namun, Anda dapat menyesuaikan nilai untuk membuat template berfungsi untuk kasus penggunaan Anda.

Di mana saya menerima data saya?

Grup titik akhir aliran data diatur untuk menggunakan antarmuka jaringan instance penerima yang dibuat oleh bagian dari template. Instance penerima menggunakan aplikasi titik akhir aliran data untuk menerima aliran data dari AWS Ground Station port yang ditentukan oleh titik akhir aliran data. Setelah diterima, data tersedia untuk konsumsi melalui port UDP 50000 pada adaptor loopback dari instance penerima. Untuk informasi selengkapnya tentang menyiapkan grup titik akhir aliran data, lihat AWS::GroundStation::. DataflowEndpointGroup

Satelit siaran publik menggunakan AWS Ground Station Agen (pita lebar)

Contoh ini dibangun dari analisis yang dilakukan di JPSS-1 - Public broadcast satellite (PBS) - Evaluasi bagian panduan pengguna.

Untuk melengkapi contoh ini, Anda harus mengasumsikan skenario -- Anda ingin menangkap jalur komunikasi HRD sebagai frekuensi menengah digital pita lebar (DiGIF) dan memprosesnya seperti yang diterima oleh Agen AWS Ground Station pada EC2 instance Amazon menggunakan SDR.

> **Note**
>
> Sinyal jalur komunikasi JPSS HRD sebenarnya memiliki bandwidth 30 MHz, tetapi Anda akan mengonfigurasi konfigurasi antena-downlink untuk memperlakukannya sebagai sinyal dengan MHz bandwidth 100 sehingga dapat mengalir melalui jalur yang benar untuk diterima oleh Agen untuk contoh ini. AWS Ground Station

Jalur komunikasi

Bagian ini Rencanakan jalur komunikasi aliran data Anda mewakili memulai. Untuk contoh ini, Anda akan memerlukan bagian tambahan dalam CloudFormation template Anda yang belum digunakan dalam contoh lain, bagian Pemetaan.

> **Note**
>
> Untuk informasi selengkapnya tentang isi CloudFormation template, lihat bagian Template.

Anda akan mulai dengan menyiapkan bagian Pemetaan di CloudFormation template Anda untuk daftar AWS Ground Station awalan berdasarkan wilayah. Hal ini memungkinkan daftar awalan mudah direferensikan oleh grup keamanan EC2 instans Amazon. Untuk informasi selengkapnya tentang menggunakan daftar awalan, lihatKonfigurasi VPC dengan Agen AWS Ground Station.

```
Mappings:
  PrefixListId:
    us-east-2:
```

```
      groundstation: pl-087f83ba4f34e3bea
    us-west-2:
      groundstation: pl-0cc36273da754ebdc
    us-east-1:
      groundstation: pl-0e5696d987d033653
    eu-central-1:
      groundstation: pl-03743f81267c0a85e
    sa-east-1:
      groundstation: pl-098248765e9effc20
    ap-northeast-2:
      groundstation: pl-059b3e0b02af70e4d
    ap-southeast-1:
      groundstation: pl-0d9b804fe014a6a99
    ap-southeast-2:
      groundstation: pl-08d24302b8c4d2b73
    me-south-1:
      groundstation: pl-02781422c4c792145
    eu-west-1:
      groundstation: pl-03fa6b266557b0d4f
    eu-north-1:
      groundstation: pl-033e44023025215c0
    af-south-1:
      groundstation: pl-0382d923a9d555425
```

Untuk bagian Parameter, Anda akan menambahkan parameter berikut. Anda akan menentukan nilai untuk ini saat membuat tumpukan melalui CloudFormation konsol.

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  AZ:
    Description: "The AvailabilityZone that the resources of this stack will be created
 in. (e.g. us-east-2a)"
    Type: AWS::EC2::AvailabilityZone::Name
```

```
  ReceiverAMI:
    Description: The Ground Station Agent AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> Anda perlu membuat key pair, dan memberikan nama untuk EC2 `EC2Key` parameter Amazon. Lihat Membuat key pair untuk EC2 instans Amazon Anda.
> Selain itu, Anda harus memberikan ID AMI spesifik wilayah yang benar, saat membuat CloudFormation tumpukan. Lihat AWS Ground Station Gambar Mesin Amazon (AMIs).

Cuplikan template yang tersisa termasuk dalam bagian Resources dari template. CloudFormation

```
Resources:
  # Resources that you would like to create should be placed within the Resources
 section.
```

Mengingat skenario kami untuk mengirimkan jalur komunikasi tunggal ke EC2 instans Amazon, Anda tahu bahwa Anda akan memiliki satu jalur pengiriman sinkron. Per Pengiriman data sinkron bagian, Anda harus menyiapkan dan mengonfigurasi EC2 instans Amazon dengan AWS Ground Station Agen, dan membuat satu atau beberapa grup titik akhir aliran data. Anda akan mulai dengan terlebih dahulu menyiapkan VPC Amazon untuk Agen. AWS Ground Station

```
  ReceiverVPC:
    Type: AWS::EC2::VPC
    Properties:
      EnableDnsSupport: 'true'
      EnableDnsHostnames: 'true'
      CidrBlock: 10.0.0.0/16
      Tags:
      - Key: "Name"
        Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent VPC"
      - Key: "Description"
        Value: "VPC for EC2 instance receiving AWS Ground Station data"
```

```
  PublicSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref ReceiverVPC
      MapPublicIpOnLaunch: 'true'
      AvailabilityZone: !Ref AZ
      CidrBlock: 10.0.0.0/20
      Tags:
      - Key: "Name"
        Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent Public
Subnet"
      - Key: "Description"
        Value: "Subnet for EC2 instance receiving AWS Ground Station data"

  RouteTable:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: !Ref ReceiverVPC
      Tags:
        - Key: Name
          Value: AWS Ground Station Example - RouteTable

  RouteTableAssociation:
    Type: AWS::EC2::SubnetRouteTableAssociation
    Properties:
      RouteTableId: !Ref RouteTable
      SubnetId: !Ref PublicSubnet

  Route:
    Type: AWS::EC2::Route
    DependsOn: InternetGateway
    Properties:
      RouteTableId: !Ref RouteTable
      DestinationCidrBlock: '0.0.0.0/0'
      GatewayId: !Ref InternetGateway

  InternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: AWS Ground Station Example - Internet Gateway

  GatewayAttachment:
```

```
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      VpcId: !Ref ReceiverVPC
      InternetGatewayId: !Ref InternetGateway
```

> **Note**
>
> Untuk informasi selengkapnya tentang konfigurasi VPC yang didukung oleh AWS Ground Station Agen, lihat Persyaratan AWS Ground Station Agen - diagram VPC.

Selanjutnya, Anda akan mengatur EC2 instance Receiver Amazon.

```
 # The placement group in which your EC2 instance is placed.
 ClusterPlacementGroup:
   Type: AWS::EC2::PlacementGroup
   Properties:
     Strategy: cluster

 # This is required for the EIP if the receiver EC2 instance is in a private subnet.
 # This ENI must exist in a public subnet, be attached to the receiver and be
associated with the EIP.
 ReceiverInstanceNetworkInterface:
   Type: AWS::EC2::NetworkInterface
   Properties:
     Description: Floating network interface
     GroupSet:
       - !Ref InstanceSecurityGroup
     SubnetId: !Ref PublicSubnet

 # An EIP providing a fixed IP address for AWS Ground Station to connect to. Attach it
to the receiver instance created in the stack.
 ReceiverInstanceElasticIp:
   Type: AWS::EC2::EIP
   Properties:
     Tags:
       - Key: Name
         Value: !Join [ "-" , [ "EIP" , !Ref "AWS::StackName" ] ]

 # Attach the ENI to the EC2 instance if using a separate public subnet.
```

```
  # Requires the receiver instance to be in a public subnet (SubnetId should be the id
 of a public subnet)
  ReceiverNetworkInterfaceAttachment:
    Type: AWS::EC2::NetworkInterfaceAttachment
    Properties:
      DeleteOnTermination: false
      DeviceIndex: 1
      InstanceId: !Ref ReceiverInstance
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

  # Associate EIP with the ENI if using a separate public subnet for the ENI.
  ReceiverNetworkInterfaceElasticIpAssociation:
    Type: AWS::EC2::EIPAssociation
    Properties:
      AllocationId: !GetAtt [ReceiverInstanceElasticIp, AllocationId]
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

  # The EC2 instance that will send/receive data to/from your satellite using AWS
 Ground Station.
  ReceiverInstance:
    Type: AWS::EC2::Instance
    DependsOn: PublicSubnet
    Properties:
      DisableApiTermination: false
      IamInstanceProfile: !Ref GeneralInstanceProfile
      ImageId: !Ref ReceiverAMI
      AvailabilityZone: !Ref AZ
      InstanceType: c5.24xlarge
      KeyName: !Ref EC2Key
      Monitoring: true
      PlacementGroupName: !Ref ClusterPlacementGroup
      SecurityGroupIds:
        - Ref: InstanceSecurityGroup
      SubnetId: !Ref PublicSubnet
      Tags:
        - Key: Name
          Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
      # agentCpuCores list in the AGENT_CONFIG below defines the cores that the AWS
 Ground Station Agent is allowed to run on. This list can be changed to suit your use-
case, however if the agent isn't supplied with enough cores data loss may occur.
      UserData:
        Fn::Base64:
          Fn::Sub:
            - |
```

```
              #!/bin/bash
              yum -y update

              AGENT_CONFIG_PATH="/opt/aws/groundstation/etc/aws-gs-agent-config.json"
              cat << AGENT_CONFIG > "$AGENT_CONFIG_PATH"
              {
                "capabilities": [
                  "arn:aws:groundstation:${AWS::Region}:${AWS::AccountId}:dataflow-
endpoint-group/${DataflowEndpointGroupId}"
                ],
                "device": {
                  "privateIps": [
                    "127.0.0.1"
                  ],
                  "publicIps": [
                    "${EIP}"
                  ],
                  "agentCpuCores": [

 24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,72,73,74,75,76,77,78,79,80,81,8
                  ]
                }
              }
              AGENT_CONFIG

              systemctl start aws-groundstation-agent
              systemctl enable aws-groundstation-agent

              # <Tuning Section Start>
              # Visit the AWS Ground Station Agent Documentation in the User Guide for
 more details and guidance updates

              # Set IRQ affinity with list of CPU cores and Receive Side Scaling mask
              # Core list should be the first two cores (and hyperthreads) on each
 socket
              # Mask set to everything currently
              # https://github.com/torvalds/linux/blob/v4.11/Documentation/networking/
scaling.txt#L80-L96
              echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh '0 1 48
 49' 'ffffffff,ffffffff,ffffffff' >>/var/log/user-data.log 2>&1" >>/var/spool/cron/root

              # Reserving the port range defined in the GS agent ingress address in
 the Dataflow Endpoint Group so the kernel doesn't steal any of them from the GS agent.
 These ports are the ports that the GS agent will ingress data
```

```
              # across, so if the kernel steals one it could cause problems ingressing
data onto the instance.
              echo net.ipv4.ip_local_reserved_ports="42000-50000" >> /etc/sysctl.conf

              # </Tuning Section End>

              # We have to reboot for linux kernel settings to apply
              shutdown -r now

            - DataflowEndpointGroupId: !Ref DataflowEndpointGroup
              EIP: !Ref ReceiverInstanceElasticIp
```

```
 # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
 # Station will use to send/receive data to/from your satellite.
 DataflowEndpointGroup:
   Type: AWS::GroundStation::DataflowEndpointGroup
   Properties:
     ContactPostPassDurationSeconds: 180
     ContactPrePassDurationSeconds: 120
     EndpointDetails:
       - AwsGroundStationAgentEndpoint:
           Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
           EgressAddress:
             SocketAddress:
               Name: 127.0.0.1
               Port: 55000
           IngressAddress:
             SocketAddress:
               Name: !Ref ReceiverInstanceElasticIp
               PortRange:
                 Minimum: 42000
                 Maximum: 55000
```

Anda juga memerlukan kebijakan, peran, dan profil yang sesuai AWS Ground Station untuk memungkinkan pembuatan elastic network interface (ENI) di akun Anda.

```
 # The security group for your EC2 instance.
 InstanceSecurityGroup:
```

```
    Type: AWS::EC2::SecurityGroup
    Properties:
      GroupDescription: AWS Ground Station receiver instance security group.
      VpcId: !Ref ReceiverVPC
      SecurityGroupEgress:
        - CidrIp: 0.0.0.0/0
          Description: Allow all outbound traffic by default
          IpProtocol: "-1"
      SecurityGroupIngress:
        # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
        - IpProtocol: udp
          Description: Allow AWS Ground Station Incoming Dataflows
          ToPort: 50000
          FromPort: 42000
          SourcePrefixListId:
            Fn::FindInMap:
              - PrefixListId
              - Ref: AWS::Region
              - groundstation

  # The EC2 instance assumes this role.
 InstanceRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Principal:
              Service:
                - "ec2.amazonaws.com"
            Action:
              - "sts:AssumeRole"
      Path: "/"
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
        - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
        - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
        - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM
        - arn:aws:iam::aws:policy/AWSGroundStationAgentInstancePolicy
      Policies:
        - PolicyDocument:
            Statement:
```

```
              - Action:
                  - sts:AssumeRole
                Effect: Allow
                Resource: !GetAtt GroundStationKmsKeyRole.Arn
            Version: "2012-10-17"
          PolicyName: InstanceGroundStationApiAccessPolicy

  # The instance profile for your EC2 instance.
  GeneralInstanceProfile:
    Type: AWS::IAM::InstanceProfile
    Properties:
      Roles:
        - !Ref InstanceRole

  # The IAM role that AWS Ground Station will assume to access and use the KMS Key for
 data delivery
  GroundStationKmsKeyRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Statement:
          - Action: sts:AssumeRole
            Effect: Allow
            Principal:
              Service:
                - groundstation.amazonaws.com
            Condition:
              StringEquals:
                "aws:SourceAccount": !Ref AWS::AccountId
              ArnLike:
                "aws:SourceArn": !Sub "arn:${AWS::Partition}:groundstation:
${AWS::Region}:${AWS::AccountId}:mission-profile/*"
          - Action: sts:AssumeRole
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"

  GroundStationKmsKeyAccessPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Statement:
          - Action:
              - kms:Decrypt
```

```
            Effect: Allow
            Resource: !GetAtt GroundStationDataDeliveryKmsKey.Arn
      PolicyName: GroundStationKmsKeyAccessPolicy
      Roles:
        - Ref: GroundStationKmsKeyRole

  GroundStationDataDeliveryKmsKey:
    Type: AWS::KMS::Key
    Properties:
      KeyPolicy:
        Statement:
          - Action:
              - kms:CreateAlias
              - kms:Describe*
              - kms:Enable*
              - kms:List*
              - kms:Put*
              - kms:Update*
              - kms:Revoke*
              - kms:Disable*
              - kms:Get*
              - kms:Delete*
              - kms:ScheduleKeyDeletion
              - kms:CancelKeyDeletion
              - kms:GenerateDataKey
              - kms:TagResource
              - kms:UntagResource
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"
            Resource: "*"
          - Action:
              - kms:Decrypt
              - kms:GenerateDataKeyWithoutPlaintext
            Effect: Allow
            Principal:
              AWS: !GetAtt GroundStationKmsKeyRole.Arn
            Resource: "*"
            Condition:
              StringEquals:
                "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
              ArnLike:
                "kms:EncryptionContext:sourceArn": !Sub "arn:
${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
```

```
          - Action:
              - kms:CreateGrant
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"
            Resource: "*"
            Condition:
              ForAllValues:StringEquals:
                "kms:GrantOperations":
                  - Decrypt
                  - GenerateDataKeyWithoutPlaintext
                "kms:EncryptionContextKeys":
                  - sourceArn
                  - sourceAccount
              ArnLike:
                "kms:EncryptionContext:sourceArn": !Sub "arn:
${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
              StringEquals:
                "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
        Version: "2012-10-17"
      EnableKeyRotation: true
```

AWS Ground Station konfigurasi

Bagian ini Buat konfigurasi mewakili memulai.

Anda memerlukan konfigurasi pelacakan untuk mengatur preferensi Anda menggunakan autotrack. Memilih PREFERRED sebagai autotrack dapat meningkatkan kualitas sinyal, tetapi tidak diperlukan untuk memenuhi kualitas sinyal karena kualitas ephemeris JPSS-1 yang memadai.

```
  TrackingConfig:
    Type: AWS::GroundStation::Config
    Properties:
      Name: "JPSS Tracking Config"
      ConfigData:
        TrackingConfig:
          Autotrack: "PREFERRED"
```

Berdasarkan jalur komunikasi, Anda harus menentukan konfigurasi antenna-downlink untuk mewakili bagian satelit, serta konfigurasi dataflow-endpoint untuk merujuk ke grup titik akhir aliran data yang mendefinisikan detail titik akhir.

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 SnppJpssDownlinkDigIfAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "SNPP JPSS Downlink WBDigIF Antenna Config"
     ConfigData:
       AntennaDownlinkConfig:
         SpectrumConfig:
           Bandwidth:
             Units: "MHz"
             Value: 100
           CenterFrequency:
             Units: "MHz"
             Value: 7812
           Polarization: "RIGHT_HAND"

 # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
 # from your satellite.
 DownlinkDigIfEndpointConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "Aqua SNPP JPSS Terra Downlink DigIF Endpoint Config"
     ConfigData:
       DataflowEndpointConfig:
         DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
         DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station profil misi

Bagian ini Buat profil misi mewakili memulai.

Sekarang setelah Anda memiliki konfigurasi terkait, Anda dapat menggunakannya untuk membangun aliran data. Anda akan menggunakan default untuk parameter yang tersisa.

```
 # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
 # uplink and downlink data to your satellite.
 SnppJpssMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: !Sub 'JPSS WBDigIF gs-agent EC2 Delivery'
     ContactPrePassDurationSeconds: 120
     ContactPostPassDurationSeconds: 120
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Ref SnppJpssDownlinkDigIfAntennaConfig
         Destination: !Ref DownlinkDigIfEndpointConfig
     StreamsKmsKey:
       KmsKeyArn: !GetAtt GroundStationDataDeliveryKmsKey.Arn
     StreamsKmsRole: !GetAtt GroundStationKmsKeyRole.Arn
```

Menyatukannya

Dengan sumber daya di atas, Anda sekarang memiliki kemampuan untuk menjadwalkan kontak JPSS-1 untuk pengiriman data sinkron dari salah satu onboard Anda. AWS Ground Station AWS Ground Station Lokasi

Berikut ini adalah CloudFormation template lengkap yang mencakup semua sumber daya yang dijelaskan dalam bagian ini digabungkan menjadi satu template yang dapat langsung digunakan CloudFormation.

CloudFormation Template bernama `DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml` dirancang untuk memberi Anda akses cepat untuk mulai menerima data frekuensi menengah digital (DiGIF) untuk satelit Aqua, SNPP, JPSS-1/NOAA-20, dan Terra. Ini berisi EC2 instans Amazon dan CloudFormation sumber daya yang diperlukan untuk menerima data siaran langsung DiGIF mentah menggunakan AWS Ground Station Agen.

Jika Aqua, SNPP, JPSS-1/NOAA-20, dan Terra tidak masuk ke akun Anda, lihat. Satelit onboard

> **Note**
>
> Anda dapat mengakses template dengan mengakses bucket Amazon S3 orientasi pelanggan menggunakan kredensi yang valid. AWS Tautan di bawah ini menggunakan bucket Amazon S3 regional. Ubah kode `us-west-2` wilayah untuk mewakili wilayah yang sesuai tempat Anda ingin membuat CloudFormation tumpukan.
> Selain itu, petunjuk berikut menggunakan YAMAL. Namun, template tersedia dalam format YAMAL dan JSON. Untuk menggunakan JSON, ganti ekstensi `.yml` file dengan `.json` saat mengunduh templat.

Untuk mengunduh templat menggunakan AWS CLI, gunakan perintah berikut:

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/agent/ec2_delivery/
DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml .
```

Anda dapat melihat dan mengunduh templat di konsol dengan menavigasi ke URL berikut di browser Anda:

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-
west-2/agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

Anda dapat menentukan template secara langsung CloudFormation menggunakan link berikut:

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

Sumber daya tambahan apa yang didefinisikan template?

`DirectBroadcastSatelliteWbDigIfEc2DataDelivery`Template mencakup sumber daya tambahan berikut:

- Receiver Instance Elastic Network Interface - (Bersyarat) Sebuah elastic network interface dibuat dalam subnet yang ditentukan oleh PublicSubnetIdjika disediakan. Ini diperlukan jika instance penerima berada di subnet pribadi. Elastic network interface akan dikaitkan dengan EIP dan dilampirkan ke instance receiver.
- Receiver Instance Elastic IP - IP elastis yang AWS Ground Station akan terhubung ke. Ini melekat pada instance receiver atau elastic network interface.

- Salah satu asosiasi IP Elastis berikut:
  - Instance Penerima ke Asosiasi IP Elastis - Asosiasi IP Elastis ke instance penerima Anda, jika tidak PublicSubnetIdditentukan. Ini membutuhkan SubnetIdreferensi subnet publik.
  - Receiver Instance Elastic Network Interface to Elastic IP Association - Asosiasi IP elastis ke instance receiver elastic network interface, jika PublicSubnetIdditentukan.
- (Opsional) Pemicu CloudWatch Acara - AWS Lambda Fungsi yang dipicu menggunakan CloudWatch Peristiwa yang dikirim oleh AWS Ground Station sebelum dan sesudah kontak. AWS Lambda Fungsi akan memulai dan secara opsional menghentikan Instance Penerima Anda.
- (Opsional) EC2 Verifikasi Amazon untuk Kontak - Opsi untuk menggunakan Lambda untuk menyiapkan sistem verifikasi EC2 instans Amazon Anda untuk kontak dengan notifikasi SNS. Penting untuk dicatat bahwa ini mungkin dikenakan biaya tergantung pada penggunaan Anda saat ini.
- Profil misi tambahan - Profil misi untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).
- Konfigurasi antena-downlink tambahan - Konfigurasi downlink antena untuk satelit siaran publik tambahan (Aqua, SNPP, dan Terra).

Nilai dan parameter untuk satelit dalam template ini sudah terisi. Parameter ini memudahkan Anda untuk AWS Ground Station segera menggunakan satelit ini. Anda tidak perlu mengkonfigurasi nilai Anda sendiri untuk digunakan AWS Ground Station saat menggunakan template ini. Namun, Anda dapat menyesuaikan nilai untuk membuat template berfungsi untuk kasus penggunaan Anda.

Di mana saya menerima data saya?

Grup titik akhir aliran data diatur untuk menggunakan antarmuka jaringan instance penerima yang dibuat oleh bagian dari template. Instance penerima menggunakan AWS Ground Station Agen untuk menerima aliran data dari AWS Ground Station port yang ditentukan oleh titik akhir aliran data. Untuk informasi selengkapnya tentang menyiapkan grup titik akhir aliran data, lihat. AWS::GroundStation::DataflowEndpointGroup Untuk informasi lebih lanjut tentang AWS Ground Station Agen, lihat Apa itu AWS Ground Station Agen?

Pemecahan Masalah

Dokumentasi berikut dapat membantu Anda memecahkan masalah yang mungkin terjadi saat menggunakan. AWS Ground Station

Topik

Memecahkan masalah kontak yang mengirimkan data ke Amazon EC2

Jika Anda tidak berhasil menyelesaikan AWS Ground Station kontak, Anda harus memverifikasi bahwa instans Amazon EC2 Anda berjalan, memverifikasi bahwa aplikasi titik akhir aliran data Anda berjalan, dan memverifikasi bahwa aliran aplikasi titik akhir aliran data Anda dikonfigurasi dengan benar.

> **Note**
>
> DataDefender (DDX) adalah contoh aplikasi endpoint aliran data yang saat ini didukung oleh AWS Ground Station

Prasyarat

Prosedur berikut mengasumsikan bahwa instans Amazon EC2 sudah disiapkan. Untuk menyiapkan instans Amazon EC2 di AWS Ground Station, lihat Memulai.

Langkah 1: Verifikasi bahwa instans EC2 Anda sedang berjalan

Prosedur berikut menunjukkan cara menemukan instans Amazon EC2 Anda di konsol dan memulainya jika tidak berjalan.

1. Temukan instans Amazon EC2 yang digunakan untuk kontak yang sedang Anda atasi masalah. Gunakan langkah-langkah berikut:

   a. Di CloudFormationdasbor Anda, pilih tumpukan yang berisi instans Amazon EC2 Anda.

   b. Pilih tab Sumber Daya dan temukan instans Amazon EC2 Anda di kolom Logical ID. Verifikasi bahwa instance dibuat di kolom Status.

   c. Di kolom ID Fisik, pilih tautan untuk instans Amazon EC2 Anda. Ini akan membawa Anda ke konsol manajemen Amazon EC2.

2. Di konsol manajemen Amazon EC2, pastikan Status Instans Amazon EC2 Anda berjalan.

3. Jika instans Anda berjalan, lanjutkan ke langkah berikutnya. Jika instans Anda tidak berjalan, mulai instance dengan menggunakan langkah berikut:

   - Dengan memilih instans Amazon EC2, pilih Tindakan > Status Instans > Mulai.

Langkah 2: Tentukan jenis aplikasi aliran data yang digunakan

Jika Anda menggunakan AWS Ground Station Agen untuk pengiriman data, silakan alihkan ke bagian Agen Pemecahan Masalah AWS Ground Station. Jika tidak, jika Anda menggunakan aplikasi DataDefender (DDX) terusthe section called “Langkah 3: Verifikasi bahwa aplikasi aliran data sedang berjalan”.

Langkah 3: Verifikasi bahwa aplikasi aliran data sedang berjalan

Memverifikasi status DataDefender mengharuskan Anda untuk terhubung ke instans Anda di Amazon EC2. Untuk detail selengkapnya tentang menghubungkan ke instans Anda, lihat Connect to your Linux instance.

Prosedur berikut menyediakan langkah-langkah pemecahan masalah menggunakan perintah dalam klien SSH.

1. Buka terminal atau prompt perintah dan sambungkan ke instans Amazon EC2 Anda menggunakan SSH. Teruskan port 80 dari host jarak jauh untuk melihat UI DataDefender web.

Perintah berikut menunjukkan cara menggunakan SSH untuk terhubung ke instans Amazon EC2 melalui benteng dengan port forwarding diaktifkan.

> **Note**
>
> Anda harus mengganti<SSH KEY>,<BASTION HOST>, dan <HOST>dengan kunci ssh spesifik, nama host bastion, dan nama host instans Amazon EC2.

Untuk Windows

```
ssh -L 8080:localhost:80 -o ProxyCommand="C:\Windows\System32\OpenSSH\ssh.exe -o
 \"ForwardAgent yes\" -W %h:%p -i \"<SSH KEY>\" ec2-user@<BASTION HOST>" -i "<SSH
 KEY>" ec2-user@<HOST>
```

Untuk Mac

```
ssh -L 8080:localhost:80 -o ProxyCommand="ssh -A -o 'ForwardAgent yes' -W %h:%p -i
 <SSH KEY> ec2-user@<BASTION HOST>" -i <SSH  KEY> ec2-user@<HOST>
```

2. Verifikasi bahwa DataDefender (juga disebut DDX) berjalan dengan mengambil (memeriksa) untuk proses yang berjalan bernama ddx dalam output. Perintah untuk grepping (memeriksa) untuk proses yang berjalan dan output contoh yang berhasil disediakan di bawah ini.

```
[ec2-user@Receiver-Instance ~]$ ps -ef | grep ddx
          Rtlogic   4977      1 10 Oct16 ?        2-00:22:14 /opt/rtlogic/ddx/
bin/ddx -m/opt/rtlogic/ddx/modules -p/opt/rtlogic/ddx/plugins -c/opt/rtlogic/
ddx/bin/ddx.xml -umask=077 -daemon -f installed=true -f security=true -f enable
 HttpsForwarding=true
          Ec2-user 18787 18657  0 16:51 pts/0      00:00:00 grep -color=auto ddx
```

Jika DataDefender sedang berjalan, lewati ke the section called “Langkah 4: Verifikasi bahwa aliran aplikasi aliran data Anda dikonfigurasi” Jika tidak, lanjutkan ke langkah berikutnya.

3. Mulai DataDefender gunakan perintah di bawah ini.

```
sudo systemctl start rtlogic-ddx
```

Jika DataDefender berjalan setelah menggunakan perintah, lompat kethe section called "Langkah 4: Verifikasi bahwa aliran aplikasi aliran data Anda dikonfigurasi". Jika tidak, lanjutkan ke langkah berikutnya.

4. Periksa file berikut menggunakan perintah di bawah ini untuk melihat apakah ada kesalahan saat menginstal dan mengkonfigurasi DataDefender.

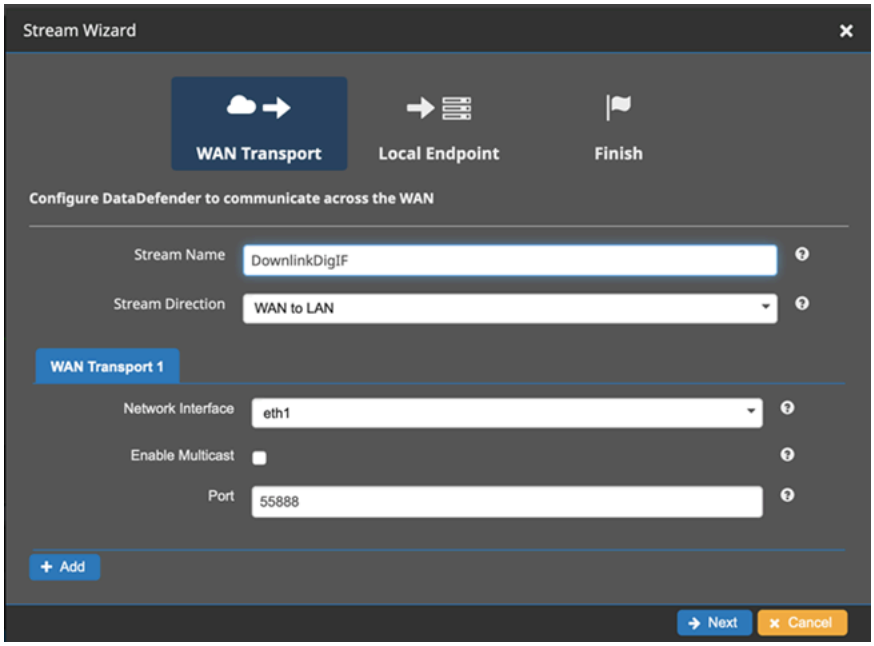
```
cat /var/log/user-data.log
          cat /opt/aws/groundstation/.startup.out
```

> **Note**
>
> Masalah umum yang ditemukan saat memeriksa file-file ini adalah bahwa VPC Amazon tempat instans Amazon EC2 Anda berjalan tidak memiliki akses ke Amazon S3 untuk mengunduh file instalasi. Jika Anda menemukan di log Anda bahwa ini adalah masalahnya, periksa pengaturan Amazon VPC dan grup keamanan instans EC2 Anda untuk memastikan mereka tidak memblokir akses ke Amazon S3.

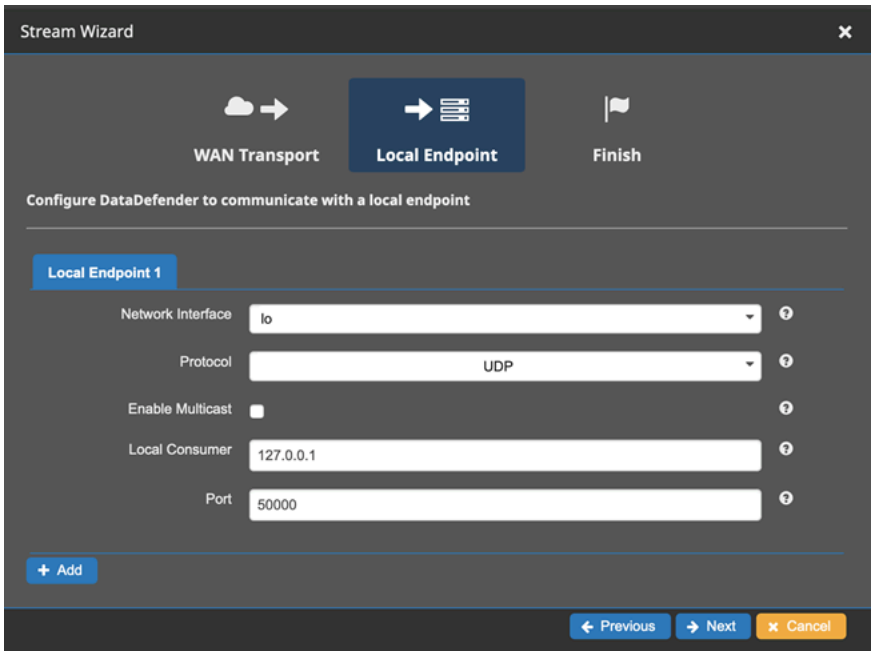
Jika DataDefender berjalan setelah memeriksa pengaturan VPC Amazon Anda, lanjutkan ke. the section called "Langkah 4: Verifikasi bahwa aliran aplikasi aliran data Anda dikonfigurasi" Jika masalah berlanjut, hubungi AWS Support dan kirim file log Anda dengan deskripsi masalah Anda.

Langkah 4: Verifikasi bahwa aliran aplikasi aliran data Anda dikonfigurasi

1. Di browser web, akses antarmuka pengguna DataDefender web Anda dengan memasukkan alamat berikut di bilah alamat: localhost:8080. Kemudian, tekan Enter.
2. Di DataDefenderdasbor, pilih Buka Detail.
3. Pilih aliran Anda dari daftar aliran, dan pilih Edit Stream.
4. Di kotak dialog Stream Wizard, lakukan hal berikut:
   a. Di panel Transportasi WAN, pastikan WAN ke LAN dipilih untuk Arah Aliran.
   b. Di kotak Port, pastikan port WAN yang Anda pilih untuk grup endpoint aliran data Anda ada. Secara default, port ini adalah 55888. Lalu, pilih Selanjutnya.



c. Di panel Endpoint Lokal, pastikan port yang valid ada di kotak Port. Secara default, port ini adalah 50000. Ini adalah port tempat Anda akan menerima data Anda DataDefender setelah menerimanya dari AWS Ground Station layanan. Lalu, pilih Selanjutnya.



d. Pilih Selesai di menu yang tersisa jika Anda telah mengubah nilai apa pun. Jika tidak, Anda dapat membatalkan menu Stream Wizard.

Anda sekarang telah memastikan bahwa instans Amazon EC2 Anda dan keduanya berjalan DataDefender dan dikonfigurasi dengan benar untuk menerima data dari.AWS Ground Station Lanjutkan ke the section called “Langkah 5: Pastikan Anda memiliki cukup alamat IP yang tersedia di subnet instance penerima Anda”.

Langkah 5: Pastikan Anda memiliki cukup alamat IP yang tersedia di subnet instance penerima Anda

Prosedur berikut menunjukkan cara menemukan jumlah alamat IP yang tersedia di instans penerima Amazon EC2 di konsol.

1. Untuk setiap instans penerima Amazon EC2 yang digunakan untuk kontak yang sedang Anda atasi masalah. Gunakan langkah-langkah berikut:
   a. Di CloudFormationdasbor Anda, pilih tumpukan yang berisi instans Amazon EC2 Anda.
   b. Pilih tab Sumber Daya dan temukan instans Amazon EC2 Anda di kolom Logical ID. Verifikasi bahwa instance dibuat di kolom Status.
   c. Di kolom ID Fisik, pilih tautan untuk instans Amazon EC2 Anda. Ini akan membawa Anda ke konsol manajemen Amazon EC2.
2. Di konsol manajemen Amazon EC2, temukan dan klik tautan Subnet ID di Ringkasan Instans instans penerima Amazon EC2 Anda. Ini akan membawa Anda ke konsol manajemen VPC Amazon yang sesuai.
3. Pilih subnet yang cocok di konsol manajemen VPC Amazon dan periksa Detail subnet Anda untuk alamat IPv4 yang Tersedia. Jika nomor ini setidaknya tidak sebanyak titik akhir aliran data yang menggunakan instance penerima Amazon EC2 ini, lakukan hal berikut:
   a. Perbarui subnet sesuai CloudFormation template Anda CidrBlockagar berukuran benar. Untuk detail lebih lanjut tentang ukuran subnet lihat, subnet CIDR memblokir.
   b. Tempatkan kembali tumpukan Anda dengan template yang diperbarui CloudFormation.

Jika Anda terus mengalami masalah, hubungi AWS Support.

Memecahkan masalah kontak GAGAL

Kontak akan memiliki status kontak terminal FAILED saat AWS Ground Station mendeteksi masalah dengan konfigurasi sumber daya Anda. Kasus penggunaan umum yang dapat menyebabkan kontak

GAGAL disediakan di bawah ini, bersama dengan langkah-langkah untuk membantu memecahkan masalah.

> **Note**
>
> Panduan ini khusus untuk status kontak GAGAL - dan tidak ditujukan untuk status kegagalan lainnya, seperti,, atau AWS_FAILEDAWS_CANCELLEDFAILED_TO_SCHEDULE. Untuk informasi selengkapnya tentang status kontak, lihat the section called “AWS Ground Station status kontak”

Kasus penggunaan gagal titik akhir Dataflow

Berikut ini adalah daftar kasus penggunaan umum yang dapat mengakibatkan status kontak GAGAL untuk aliran data berbasis titik akhir aliran data:

- Titik akhir Dataflow tidak pernah terhubung - Koneksi antara AWS Ground Station Antena dan Grup Titik Akhir Dataflow Anda untuk satu atau lebih aliran data tidak pernah dibuat.
- Titik akhir Dataflow terhubung terlambat - Koneksi antara AWS Ground Station Antena dan Grup Titik Akhir Dataflow Anda untuk satu atau lebih aliran data dibuat setelah waktu mulai kontak.
- Subnet endpoint Dataflow kehabisan alamat IP yang tersedia - AWS Ground Station solusi pengiriman data tidak dapat membuat ENI di jaringan pribadi Anda karena tidak memiliki alamat IP yang tersedia di subnet instance penerima.
- Subnet endpoint Dataflow tidak valid - AWS Ground Station solusi pengiriman data tidak dapat membuat ENI di jaringan pribadi Anda karena ketidakmampuan untuk mengakses subnet yang disediakan yang ditentukan dalam Dataflow Endpoint Group.

Untuk kasus kegagalan titik akhir aliran data apa pun, disarankan untuk melihat hal-hal berikut:

- Konfirmasikan instans Amazon EC2 penerima berhasil dimulai, sebelum waktu mulai kontak.
- Konfirmasikan perangkat lunak titik akhir aliran data aktif dan berjalan selama kontak.
- Pastikan Anda memiliki setidaknya satu alamat IP yang tersedia per titik akhir aliran data per subnet instance penerima.
- Pastikan subnet yang terkait dengan Grup Titik Akhir Dataflow Anda, melalui aliran data yang dikonfigurasi, tetap aktif dan tersedia Siapkan dan konfigurasikan Amazon VPC untuk. AWS Ground Station

Lihat bagian Memecahkan masalah kontak yang mengirimkan data ke Amazon EC2 untuk langkah-langkah pemecahan masalah yang lebih spesifik.

AWS Ground Station Kasus penggunaan agen GAGAL

Berikut ini adalah daftar kasus penggunaan umum yang dapat mengakibatkan status kontak GAGAL untuk aliran data berbasis Agen:

- AWS Ground Station Status Agen Tidak Pernah Dilaporkan - Agen yang bertanggung jawab untuk mengatur pengiriman data di Grup Titik Akhir Dataflow Anda untuk satu atau lebih aliran data yang tidak pernah berhasil melaporkan statusnya. AWS Ground Station Pembaruan status ini akan terjadi dalam beberapa detik dari waktu akhir kontak.
- AWS Ground Station Agen Mulai Terlambat - Agen yang bertanggung jawab untuk mengatur pengiriman data di Grup Titik Akhir Dataflow Anda untuk satu atau lebih aliran data dimulai terlambat, setelah waktu mulai kontak.

Untuk kasus kegagalan aliran data AWS Ground Station Agen, disarankan untuk melihat hal-hal berikut:

- Konfirmasikan instans Amazon EC2 penerima berhasil dimulai, sebelum waktu mulai kontak.
- Konfirmasikan aplikasi Agen sudah aktif dan berjalan di awal dan selama kontak.
- Konfirmasikan aplikasi Agen dan instans Amazon EC2 tidak dimatikan dalam waktu 15 detik setelah kontak berakhir. Ini memberi Agen waktu yang cukup untuk melaporkan status ke AWS Ground Station.

Lihat bagian Memecahkan masalah kontak yang mengirimkan data ke Amazon EC2 untuk langkah-langkah pemecahan masalah yang lebih spesifik.

Memecahkan masalah pembaruan kontak yang gagal

Saat Anda memanggil UpdateContactAPI, AWS Ground Station lakukan validasi sinkron pada permintaan. Jika validasi berlalu, pembaruan diproses secara asinkron untuk menyebarkan perubahan ke wilayah antena. Kesalahan validasi sinkron dikembalikan langsung dalam respons HTTP. Kegagalan asinkron dilaporkan melalui `failureCodes` dan `failureMessage` bidang pada versi kontak, yang dapat Anda lihat dengan memanggil DescribeContactVersionversi yang gagal diperbarui.

Untuk informasi selengkapnya tentang pembuatan versi kontak, lihat. Perbarui kontak dan versi kontak

Kesalahan validasi sinkron

Kesalahan berikut dikembalikan langsung dalam respons HTTP ketika UpdateContactpermintaan gagal validasi.

ResourceNotFoundException: Kontak tidak ditemukan

Penyebab umum

Yang ditentukan `contactId` tidak ada atau milik AWS akun yang berbeda.

Resolusi

1. Verifikasi bahwa `contactId` sudah benar.
2. Konfirmasikan bahwa Anda menggunakan kredensi untuk AWS akun yang memiliki kontak.
3. Gunakan ListContactsuntuk menemukan yang benar`contactId`.

ConflictException: Tidak dapat memperbarui kontak

Penyebab umum

Kontak berada dalam keadaan yang tidak mengizinkan pembaruan. UpdateContactAPI hanya dapat dipanggil ketika kontak berada di`SCHEDULED`,`PREPASS`, atau `PASS` status. Kesalahan ini juga terjadi jika pembaruan lain sudah berlangsung (versi kontak terbaru dalam `UPDATING` status).

Resolusi

1. Hubungi DescribeContactuntuk memeriksa status kontak saat ini.
2. Jika kontak dalam status terminal (misalnya,, `COMPLETED`FAILED, atau`CANCELLED`), itu tidak dapat diperbarui. Kontak hanya dapat diperbarui ketika berada di`SCHEDULED`,`PREPASS`, atau `PASS` status. Untuk daftar lengkap status terminal, lihatAWS Ground Station status kontak.
3. Jika pembaruan lain sedang berlangsung, tunggu pembaruan saat ini mencapai `ACTIVE` atau `FAILED_TO_UPDATE` status sebelum mengirimkan pembaruan lain. Anda dapat melakukan polling DescribeContactVersionAPI, atau menggunakan utilitas kenyamanan `ContactUpdated` pelayan yang disediakan oleh beberapa orang AWS SDKs dan. AWS Command Line Interface

InvalidParameterException: Parameter permintaan tidak valid

Penyebab umum

Permintaan berisi parameter yang tidak valid. Penyebab umum meliputi:

- Hilang atau kosong`clientToken`.
- Beberapa jenis `ProgramTrackSettings` (azimuth/elevasi, OEM, dan TLE) termasuk dalam satu permintaan. Hanya satu jenis yang diizinkan per permintaan.
- Pengaturan `satelliteArn` ke nol tanpa disetujui untuk ephemeris elevasi azimuth di stasiun bumi kontak.
- Hilang `AzElProgramTrackSettings` `satelliteArn` saat nol.
- Menyediakan `ephemerisId` yang tidak terkait dengan yang ditentukan`satelliteArn`.
- Satelit tidak memiliki jendela visibilitas yang valid dari stasiun bumi untuk rentang waktu kontak.
- Satelit tidak terpasang ke stasiun bumi atau tidak memiliki lisensi yang diperlukan oleh profil misi.
- Profil misi menyertakan Antena Downlink Demod Decode Config konfigurasi, yang tidak didukung untuk pembaruan kontak.

Resolusi

1. Tinjau pesan kesalahan dalam respons untuk detail tentang parameter mana yang tidak valid.
2. Pastikan Anda memberikan tepat satu jenis `ProgramTrackSettings` per permintaan.
3. Jika menggunakan sudut azimuth/elevation penunjuk tanpa `asatelliteArn`, konfirmasikan bahwa akun Anda disetujui untuk kemampuan ini di stasiun bumi. Untuk informasi selengkapnya, lihat Berikan data ephemeris elevasi azimuth.
4. Verifikasi bahwa ephemeris yang Anda referensikan dikaitkan dengan satelit yang benar dan mencakup rentang waktu kontak.

ResourceLimitExceededException: Batas versi maksimum tercapai

Penyebab umum

Kontak telah mencapai jumlah maksimum versi (128). Setiap panggilan untuk UpdateContactmembuat versi baru, dan kontak tidak dapat melebihi batas ini.

Resolusi

1. Batas ini tidak dapat dinaikkan. Jika Anda perlu membuat perubahan lebih lanjut, batalkan kontak dan pesan yang baru.

Kode kegagalan asinkron

Kode kegagalan berikut muncul di `failureCodes` bidang versi kontak dengan `FAILED_TO_UPDATE` status. Gunakan DescribeContactVersionuntuk mengambil detail ini. `failureMessage`Bidang ini memberikan konteks tambahan tentang kegagalan.

Kode kegagalan	Penyebab umum	Resolusi
`INTERNAL_ERROR`	Terjadi kesalahan internal yang tidak terduga saat memproses pembaruan.	Coba lagi pembaruan. Jika masalah berlanjut, hubungi AWS Dukungan.
`INVALID_SATELLITE_ARN`	ARN satelit yang disediakan dalam permintaan pembaruan tidak valid atau tidak ada.	Verifikasi ARN satelit dan konfirmasikan bahwa satelit terdaftar di akun Anda.
`INVALID_UPDATE_CONTACT_REQUEST`	Permintaan pembaruan berisi parameter tidak valid yang tidak tertangkap selama validasi sinkron.	Tinjau detailnya dan perbaiki parameter permintaan. `failureMessage`
`EPHEMERIS_NOT_FOUND`	Ephemeris yang direferensikan dalam penggantian pelacakan tidak ada.	Verifikasi `ephemerisId` dan konfirmasikan bahwa ephemeris belum dihapus.
`EPHEMERIS_TIME_RANGE_INVALID`	Ephemeris tidak mencakup rentang waktu kontak.	Unggah ephemeris baru yang mencakup rentang waktu kontak penuh. Jika rentang waktu ephemeris tidak dapat diperpanjang, batalkan kontak dan pesan yang baru selama rentang waktu ephemeris. Untuk informasi selengkapnya, lihat Berikan data ephemeris khusus.

Kode kegagalan	Penyebab umum	Resolusi
`EPHEMERIS_NOT_ENABLED`	Ephemeris yang direferensikan tidak dalam keadaan. `ENABLED`	Periksa status ephemeris dan aktifkan sebelum mencoba kembali pembaruan.
`SATELLITE_DOES_NOT_MATCH_EPHEMERIS`	Ephemeris tidak terkait dengan satelit yang ditentukan dalam permintaan pembaruan.	Pastikan `ephemerisId` milik satelit yang ditentukan dalam`satelliteArn` .
`NOT_ONBOARDED_TO_AZEL_EPHEMERIS`	Akun Anda tidak disetujui untuk menggunakan data azimuth elevation ephemeris di stasiun bumi kontak. Ephemeris elevasi Azimuth adalah fitur terbatas yang tersedia untuk sejumlah kasus penggunaan khusus.	Jika azimuth elevation ephemeris diperlukan untuk kasus penggunaan Anda, buka AWS Dukungan tiket melalui untuk meminta akses. AWS Support Center Console Atau, pertimbangkan untuk menggunakan data ephemeris TLE atau data ephemeris OEM jika sesuai dengan kasus penggunaan Anda.
`AZEL_EPHEMERIS_NOT_FOUND`	Ephemeris elevasi azimuth yang dirujuk dalam permintaan tidak ada.	Verifikasi `ephemerisId` dan konfirmasikan bahwa ephemeris elevasi azimuth belum dihapus.
`AZEL_EPHEMERIS_WRONG_GROUND_STATION`	Ephemeris elevasi azimuth dibuat untuk stasiun bumi yang berbeda dari yang digunakan oleh kontak.	Unggah ephemeris elevasi azimuth baru untuk stasiun bumi yang benar, atau gunakan ephemeris yang ada yang cocok dengan stasiun bumi kontak.
`AZEL_EPHEMERIS_INVALID_STATUS`	Ephemeris elevasi azimuth tidak dalam keadaan valid untuk digunakan.	Periksa status ephemeris. Itu harus dalam `ENABLED` keadaan. Jika ephemeris gagal validasi, unggah versi yang diperbaiki.

Kode kegagalan	Penyebab umum	Resolusi
`AZEL_EPHEMERIS_TIME_RANGE_INVALID`	Ephemeris elevasi azimuth tidak mencakup rentang waktu kontak.	Unggah ephemeris elevasi azimuth baru yang mencakup rentang waktu kontak penuh. Jika rentang waktu ephemeris tidak dapat diperpanjang, batalkan kontak dan pesan yang baru selama rentang waktu ephemeris.

Memeriksa status pembaruan

Setelah meneleponUpdateContact, versi kontak baru dimulai di UPDATING negara bagian. Selama waktu ini, DescribeContactterus kembalikan versi kontak yang sebelumnya aktif. Versi baru tidak muncul DescribeContact sampai telah disebarkan ke antena dan mencapai ACTIVE status. Untuk memeriksa status versi tertentu, gunakan DescribeContactVersion.

Untuk menentukan apakah pembaruan berhasil atau gagal:

1. Panggil DescribeContactVersiondengan contactId dan versionId dikembalikan oleh UpdateContact tanggapan.
2. Periksa version.status bidangnya. Status ACTIVE berarti pembaruan berhasil diterapkan. Status FAILED_TO_UPDATE berarti pembaruan gagal.
3. Jika statusnyaFAILED_TO_UPDATE, periksa version.failureMessage kolom version.failureCodes dan untuk detail tentang apa yang salah.

> **Tip**
>
> Beberapa AWS SDKs dan AWS Command Line Interface mendukung ContactUpdated pelayan yang secara otomatis melakukan polling DescribeContactVersion hingga versi mencapai ACTIVE atau FAILED_TO_UPDATE status. Misalnya, AWS Command Line Interface menyediakan perintah aws groundstation wait contact-updated. Gunakan pelayan alih-alih menerapkan logika polling Anda sendiri.

Memecahkan masalah kontak FAILED_TO_SCHEDULE

Kontak akan berakhir dalam status FAILED_TO_SCHEDULE saat AWS Ground Station mendeteksi masalah baik dengan konfigurasi sumber daya Anda atau dalam sistem internal. Kontak yang berakhir dengan status FAILED_TO_SCHEDULE secara opsional akan menyediakan konteks tambahan untuk. `errorMessage` Untuk informasi tentang menjelaskan kontak, lihat DescribeContactAPI.

Kasus penggunaan umum yang dapat menyebabkan kontak FAILED_TO_SCHEDULE disediakan di bawah ini, bersama dengan langkah-langkah untuk membantu memecahkan masalah.

> **Note**
>
> Panduan ini khusus untuk status kontak FAILED_TO_SCHEDULE - dan tidak ditujukan untuk status kegagalan lainnya, seperti,, atau GAGAL. AWS_FAILEDAWS_CANCELLED Untuk informasi selengkapnya tentang status kontak, lihat the section called “AWS Ground Station status kontak”

Pengaturan yang ditentukan dalam Antenna Downlink Demod Decode Config tidak didukung

Profil misi yang digunakan untuk menjadwalkan kontak ini memiliki antenna-downlink-demod-decode konfigurasi yang tidak valid.

AntennaDownlinkDemodDecode Konfigurasi yang sudah ada sebelumnya

- Jika antenna-downlink-demod-decode konfigurasi Anda baru saja diubah - putar kembali ke versi yang sebelumnya berfungsi sebelum mencoba menjadwalkan.
- Jika ini adalah perubahan yang disengaja pada konfigurasi yang ada, atau konfigurasi yang sudah ada sebelumnya yang tidak lagi berhasil menjadwalkan - ikuti langkah berikutnya tentang cara mengaktifkan konfigurasi baru. AntennaDownlinkDemodDecode

AntennaDownlinkDemodDecode Konfigurasi yang baru dibuat

Hubungi AWS Ground Station langsung ke onboard konfigurasi baru Anda. Buat kasus dengan AWS Support termasuk `contactId` yang diakhiri dengan status FAILED_TO_SCHEDULE

Langkah Pemecahan Masalah Umum

Jika langkah pemecahan masalah sebelumnya tidak menyelesaikan masalah Anda:

- Coba kembali penjadwalan kontak atau jadwalkan kontak lain menggunakan profil misi yang sama. Untuk informasi tentang cara memesan kontak, lihat ReserveContact.
- Jika Anda terus menerima status FAILED_TO_SCHEDULE untuk profil misi ini, hubungi AWS Support

Memecahkan masalah DataflowEndpointGroups tidak dalam keadaan SEHAT

Di bawah ini adalah alasan grup titik akhir aliran data Anda mungkin tidak dalam `HEALTHY` keadaan serta tindakan korektif yang tepat untuk diambil.

- `NO_REGISTERED_AGENT`- Mulai instans EC2 Anda, yang akan mendaftarkan agen. Perhatikan bahwa Anda harus memiliki file konfigurasi pengontrol yang valid agar panggilan ini berhasil. Lihat Gunakan AWS Ground Station Agen untuk detail tentang mengonfigurasi file itu.
- `INVALID_IP_OWNERSHIP`- Gunakan DeleteDataflowEndpointGroup API untuk menghapus Dataflow Endpoint Group, lalu gunakan CreateDataflowEndpointGroup API untuk membuat ulang Dataflow Endpoint Group menggunakan alamat IP dan port yang terkait dengan instans EC2.
- `UNVERIFIED_IP_OWNERSHIP`- Alamat IP belum divalidasi. Validasi terjadi secara berkala sehingga ini harus diselesaikan sendiri.
- `NOT_AUTHORIZED_TO_CREATE_SLR`- Akun tidak berwenang untuk membuat Peran Tertaut Layanan yang diperlukan. Periksa langkah-langkah pemecahan masalah di Gunakan peran terkait layanan untuk Ground Station

Memecahkan masalah ephemerides yang tidak valid

Saat Anda mengunggah data ephemeris ke AWS Ground Station, data tersebut akan melalui alur kerja validasi asinkron. Jika validasi gagal, status ephemeris akan berubah menjadi. `INVALID` Pesan kesalahan dalam DescribeEphemerisrespons memberikan informasi terperinci untuk membantu Anda mengidentifikasi dan menyelesaikan masalah.

Memahami kesalahan validasi ephemeris

Ketika ephemeris gagal validasi, respons DescribeEphemerisAPI mencakup dua bidang untuk membantu mendiagnosis masalah:

errorCode

Kode yang dapat dibaca mesin yang mengidentifikasi kesalahan validasi spesifik. Ini dapat digunakan untuk penanganan kesalahan terprogram.

errorMessage

Deskripsi kesalahan validasi yang dapat dibaca manusia dengan detail spesifik tentang apa yang salah dan panduan tentang cara memperbaikinya.

Contoh DescribeEphemerisrespons untuk ephemeris yang tidak valid:

```
{
    "ephemerisId": "abc12345-6789-def0-1234-567890abcdef",
    "name": "My Invalid Ephemeris",
    "status": "INVALID",
    "creationTime": 1620254718.765,
    "invalidReason": "METADATA_INVALID",
    "errorCode": "OBJECT_NAME_MISSING",
    "errorMessage": "Metadata field missing: OBJECT_NAME",
    "suppliedData": {
        "tle": {
            "ephemerisData": "[...]"
        }
    }
}
```

Kesalahan validasi umum untuk ephemerides TLE

Berikut ini adalah kesalahan validasi umum yang ditemui saat mengunggah ephemerides TLE:

Nomor katalog satelit yang tidak cocok

Kesalahan: “Nomor katalog satelit yang ada di ephemeris tidak cocok dengan nomor katalog satelit satelit terkait”

Solusi: Verifikasi bahwa nomor ID/satellite katalog NORAD di baris TLE Anda cocok dengan nomor katalog satelit satelit satelit Anda. Gunakan `00000` untuk satelit tanpa nomor katalog yang ditetapkan.

Gerakan rata-rata tidak valid

Kesalahan: “Gerakan rata-rata dari ephemeris yang disediakan terlalu berbeda dari ephemeris referensi terbaru”

Solusi: Verifikasi bahwa data TLE Anda benar dan mewakili orbit yang valid. Ground Station menggunakan ephemerides Space-Track sebagai referensi selama validasi.

Kesalahan validasi umum untuk ephemerides OEM

Berikut ini adalah kesalahan validasi umum yang ditemui saat mengunggah ephemerides OEM:

Kerangka referensi tidak valid

Kesalahan: “REF_FRAME tidak didukung”

Solusi: Perbarui file OEM Anda untuk menggunakan salah satu kerangka referensi yang didukung: EME2000 atau ITRF2000.

Kolom wajib tidak ada

Kesalahan: “Bidang metadata hilang: INTERPOLASI”

Solusi: Tambahkan bidang INTERPOLASI dan INTERPOLATION_DEGREE ke bagian metadata OEM Anda. Ini diperlukan AWS Ground Station untuk menghasilkan sudut penunjuk antena yang akurat.

Sistem waktu yang tidak didukung

Kesalahan: “TIME_SYSTEM tidak didukung”

Solusi: Pastikan file OEM Anda menggunakan UTC sebagai sistem waktu.

Versi OEM yang tidak didukung

Kesalahan: “CCSDS_OEM_VERS tidak didukung”

Solusi: Pastikan file OEM Anda menggunakan CCSDS OEM versi 2.0.

Kesalahan validasi umum untuk ephemerides elevasi azimuth

Berikut ini adalah kesalahan validasi umum yang ditemui saat mengunggah ephemerides elevasi azimuth:

azimuth/elevation Data hilang

Kesalahan: “Tidak ada TimeAzEl bidang yang ada di setidaknya satu AzElSegment”

Solusi: Pastikan setiap segmen dalam data elevasi azimuth Anda berisi setidaknya satu pasangan yang diberi tag waktu. azimuth/elevation

Rentang sudut azimuth (derajat) tidak valid

Kesalahan: "AzEl az harus lebih besar dari atau sama dengan -180 dan kurang atau sama dengan 360 derajat”

Solusi: Verifikasi bahwa sudut azimuth berada dalam [-180, 360] derajat.

Rentang sudut elevasi (derajat) tidak valid

Kesalahan: "AzEl el harus lebih besar dari atau sama dengan -90 dan kurang dari atau sama dengan 90 derajat”

Solusi: Verifikasi bahwa sudut elevasi berada dalam [-90, 90] derajat.

Rentang sudut azimuth (radian) tidak valid

Kesalahan: "AzEl az harus lebih besar dari atau sama dengan -pi dan kurang dari atau sama dengan 2pi radian”

Solusi: Verifikasi bahwa sudut azimuth berada dalam radian [-π, 2π].

Rentang sudut elevasi tidak valid (radian)

Kesalahan: "AzEl el harus lebih besar dari atau sama dengan -pi/2 dan kurang dari atau sama dengan pi/2 radian”

Solusi: Verifikasi bahwa sudut elevasi berada dalam radian [-π/2, π/2].

Nilai waktu non-monotonik

Kesalahan: “ TimeAzEl Item di dalam a AzElSegment harus berurutan sementara”

Solusi: Pastikan bahwa nilai waktu di setiap segmen meningkat secara ketat.

Segmen rusak

Kesalahan: "AzElSegments harus secara temporal dalam urutan”

Solusi: Pastikan segmen disusun dalam urutan kronologis.

Segmen yang tumpang tindih

Kesalahan: “Rentang waktu setidaknya satu segmen tumpang tindih dengan rentang waktu segmen lainnya”

Solusi: Pastikan setiap segmen memiliki rentang waktu yang unik dan tidak tumpang tindih. Satu `endTime` segmen tidak boleh melebihi `startTime` segmen berikutnya.

Langkah pemecahan masalah

Jika ephemeris Anda gagal validasi, ikuti langkah-langkah berikut untuk mengatasi masalah:

1. Panggil DescribeEphemerisdengan ID ephemeris Anda untuk mengambil dan. `errorCode` `errorMessage`
2. Tinjau pesan galat untuk detail spesifik tentang pemeriksaan validasi yang gagal.
3. Perbaiki masalah yang diidentifikasi dalam data ephemeris Anda.
4. Unggah ephemeris baru dengan data yang dikoreksi menggunakan. CreateEphemeris
5. Pantau status ephemeris baru hingga mencapai keadaan. `ENABLED`
6. Hapus ephemeris yang tidak valid menggunakan DeleteEphemerisjika tidak lagi diperlukan.

Referensi kode kesalahan lengkap

Bagian berikut memberikan pemetaan komprehensif dari semua `errorCode` nilai yang dapat dikembalikan ketika validasi ephemeris gagal, diatur oleh kategori tingkat tinggi. `invalidReason`

Alasan Tidak Valid: **`METADATA_INVALID`**

Kesalahan ini terjadi ketika bidang metadata yang diperlukan hilang, salah diformat, atau berisi nilai yang tidak didukung dalam data ephemeris.

Kode Kesalahan	Pesan Kesalahan
MISMATCHED_SATCAT_ID	Nomor katalog satelit yang ada di ephemeris TLE tidak cocok dengan nomor katalog satelit satelit terkait
OEM_VERSION_TIDAK DIDUKUNG	`CCSDS_OEM_VERS` Dalam ephemeris OEM tidak didukung. Nilai yang didukung: [`2.0`]
ORIGINATOR_MISSING	Bidang `ORIGINATOR` header hilang dari ephemeris OEM
CREATION_DATE_MISSING	Bidang `CREATION_DATE` header hilang dari ephemeris OEM
OBJECT_NAME_MISSING	Bidang `OBJECT_NAME` metadata hilang dari ephemeris OEM
OBJECT_ID_MISSING	Bidang `OBJECT_ID` metadata hilang dari ephemeris OEM
REF_FRAME_UNSUPPORTED	`REF_FRAME` Dalam ephemeris OEM tidak didukung. Nilai yang didukung: [`EME2000`,`ITRF2000`]
REF_FRAME_EPOCH_UN SUPPORTED	Bidang `REF_FRAME_EPOCH` metadata di ephemeris OEM tidak didukung. Harap hapus bidang ini dari ephemeris
TIME_SYSTEM_UNSUPPORTED	`TIME_SYSTEM` Dalam ephemeris OEM tidak didukung. Nilai yang didukung: [`UTC`]
CENTER_BODY_UNSUPPORTED	`CENTER_BODY` Dalam ephemeris OEM tidak didukung. Nilai yang didukung: [`Earth`]
INTERPOLASI_HILANG	Bidang `INTERPOLATION` metadata hilang dari ephemeris OEM
INTERPOLATION_DEGREE_INVALI D	Tingkat interpolasi dalam ephemeris OEM harus lebih besar dari 0 untuk metode interpolasi

Kode Kesalahan	Pesan Kesalahan
AZ_EL_SEGMENT_LIST_MISSING	`azElSegmentList` Bidang hilang
INSUFFICIENT_TIME_AZ_EL	Tidak ada `TimeAzEl`bidang yang ada di setidaknya satu `azElSegmentList`

Alasan Tidak Valid: **`TIME_RANGE_INVALID`**

Kesalahan ini terjadi ketika ephemeris berisi rentang waktu yang tidak valid, termasuk masalah dengan start/end waktu, urutan segmen, segmen yang tumpang tindih, atau inkonsistensi temporal.

Kode Kesalahan	Pesan Kesalahan
START_TIME_IN_FUTURE	Waktu mulai Ephemeris ada di masa depan, tetapi harus di masa lalu
END_TIME_IN_PAST	Waktu akhir Ephemeris ada di masa lalu, tetapi harus di masa depan
KEDALUWARSA_TIME_T OO_EARLY	Waktu kedaluwarsa yang diberikan lebih awal dari waktu akhir ephemeris
START_TIME_METADAT A_TOO_EARLY	Nilai `START_TIME` metadata lebih awal dari waktu paling awal yang ada dalam data ephemeris OEM
STOP_TIME_METADATA _TOO_LATE	Nilai `STOP_TIME` metadata lebih lambat dari waktu terakhir yang ada dalam data ephemeris OEM
AZ_EL_SEGMENT_END_ TIME_BEFORE_START_TIME	Setidaknya satu segmen data sebelum segmen `endTimestartTime`
AZ_EL_SEGMENT_TIME S_TUMPANG TINDIH	Rentang waktu setidaknya satu segmen tumpang tindih dengan rentang waktu segmen lainnya
AZ_EL_SEGMENTS_OUT _OF_ORDER	Segmen tidak diurutkan secara temporal

Kode Kesalahan	Pesan Kesalahan
TIME_AZ_EL_ITEMS_OUT_OF_ORDER	`TimeAzEl`Item di dalam a `AzElSegment` harus secara temporal dalam urutan
AZ_EL_SEGMENT_REFERENCE_EPOCH_TIDAK VALID	Epoch referensi untuk segmen tidak valid atau salah diformat
AZ_EL_SEGMENT_START_TIME_TIDAK VALID	Waktu mulai dalam rentang waktu valid segmen tidak dimulai setelah segmen pertama
AZ_EL_SEGMENT_END_TIME_TIDAK VALID	Waktu akhir dalam rentang waktu valid segmen tidak berakhir setelah segmen terakhir
AZ_EL_SEGMENT_VALID_TIME_RANGE_TIDAK VALID	Rentang waktu yang valid untuk segmen tidak valid
AZ_EL_SEGMENT_END_TIME_TOO_LATE	Waktu akhir suatu segmen melebihi durasi maksimum yang diizinkan dari zaman referensi
AZ_EL_TOTAL_DURATION_TERLAMPAUI	Total durasi di semua segmen melebihi durasi sudut penunjuk maksimum yang diizinkan

Alasan Tidak Valid: **`TRAJECTORY_INVALID`**

Kesalahan ini terjadi ketika ephemeris berisi data lintasan yang tidak valid, termasuk masalah dengan parameter orbital, rentang sudut, atau unit.

Kode Kesalahan	Pesan Kesalahan
MEAN_MOTION_INVALID	Gerakan rata-rata ephemeris TLE yang disediakan sangat berbeda dari ephemeris referensi terbaru. Catatan: Ground Station menggunakan ephemerides Space-Track sebagai referensi selama validasi
TIME_AZ_EL_AZ_RADIAN_RANGE_TIDAK VALID	AzEl `az`harus lebih besar dari atau sama dengan -π dan kurang dari atau sama dengan 2π radian

Kode Kesalahan	Pesan Kesalahan
TIME_AZ_EL_EL_RADIAN_RANGE_TIDAK VALID	AzEl `el`harus lebih besar dari atau sama dengan -π/2 dan kurang dari atau sama dengan π/2 radian
TIME_AZ_EL_AZ_DEGREE_RANGE_INVALID	AzEl `az`harus lebih besar dari atau sama dengan -180 dan kurang atau sama dengan 360 derajat
TIME_AZ_EL_EL_DEGREE_RANGE_INVALID	AzEl `el`harus lebih besar dari atau sama dengan -90 derajat dan kurang dari atau sama dengan 90 derajat
TIME_AZ_EL_ANGLE_UNITS_TIDAK VALID	Satuan sudut tidak valid AzEl

Alasan Tidak Valid: **`KMS_KEY_INVALID`**

Kesalahan ini terjadi ketika ada masalah dengan kunci AWS Key Management Service (KMS) AWS Management Service yang digunakan untuk mengenkripsi data ephemeris.

Kode Kesalahan	Pesan Kesalahan
INSUFFICIENT_KMS_PERMISSIONS	Ground Station tidak memiliki izin yang cukup untuk mengakses kunci KMS ephemeris ini

Alasan Tidak Valid: **`VALIDATION_ERROR`**

Kesalahan ini terjadi ketika ada masalah validasi umum dengan data ephemeris yang tidak termasuk dalam kategori spesifik lainnya.

Kode Kesalahan	Pesan Kesalahan
INTERNAL_ERROR	Kesalahan internal terjadi selama validasi ephemeris
FILE_FORMAT_TIDAK VALID	Format file ephemeris tidak valid atau rusak. Verifikasi file sesuai dengan format yang diharapkan untuk jenis ephemeris

Memecahkan masalah kontak yang tidak menerima data

Mungkin saja kontak tampak berhasil, tetapi masih tidak menerima data apa pun. Ini mungkin berarti bahwa Anda menerima file PCAP yang kosong, atau tidak ada file PCAP sama sekali jika Anda menggunakan pengiriman data S3. Ini bisa terjadi karena sejumlah alasan. Berikut ini membahas beberapa penyebab, dan bagaimana mengatasinya.

Konfigurasi downlink salah

Setiap kontak yang menerima data dari satelit akan memiliki kontak terkait Konfigurasi Downlink Antena atauAntena Downlink Demod Decode Config. Jika konfigurasi yang ditentukan tidak sesuai dengan sinyal yang ditransmisikan oleh satelit, tidak AWS Ground Station akan dapat menerima sinyal yang ditransmisikan. Ini akan mengakibatkan tidak ada data yang diterima oleh AWS Ground Station.

Untuk memperbaikinya, harap verifikasi bahwa konfigurasi yang Anda gunakan setuju dengan sinyal yang dikirimkan oleh satelit Anda. Misalnya, verifikasi bahwa Anda telah menetapkan frekuensi pusat, bandwidth, polarisasi, dan jika diperlukan, parameter demodulasi dan decoding yang benar.

Manuver satelit

Ada kalanya satelit dapat melakukan manuver yang untuk sementara menonaktifkan beberapa sistem komunikasinya. Manuver juga dapat secara signifikan mengubah lokasi satelit di langit. AWS Ground Station tidak akan dapat menerima sinyal dari satelit yang tidak mentransmisikan sinyal, atau jika ephemeris yang digunakan menyebabkan AWS Ground Station antena menunjuk ke lokasi di langit di mana satelit tidak ada.

Jika Anda mencoba berkomunikasi dengan satelit siaran publik yang dioperasikan oleh NOAA, Anda mungkin dapat menemukan pesan yang menjelaskan pemadaman atau manuver di halaman Pesan Peringatan Satelit NOAA. Pesan dapat mencakup garis waktu kapan transmisi data diharapkan untuk dilanjutkan, atau ini dapat diposting dalam pesan berikutnya.

Jika Anda berkomunikasi dengan satelit Anda sendiri, Anda bertanggung jawab untuk memahami operasi satelit Anda, dan bagaimana hal ini dapat berdampak pada komunikasi. AWS Ground Station Jika Anda melakukan manuver yang akan memengaruhi lintasan satelit, ini mungkin termasuk menyediakan data ephemeris khusus yang diperbarui. Untuk informasi selengkapnya tentang penyediaan data ephemeris khusus, lihat. Memahami caranya AWS Ground Station menggunakan ephemerides

AWS Ground Station pemadaman

Jika AWS Ground Station menyebabkan kontak gagal, atau membatalkannya, AWS Ground Station akan mengatur status kontak ke AWS_FAILED, atau AWS_CANCELLED. Untuk informasi selengkapnya tentang siklus hidup kontak, lihat. Memahami siklus hidup kontak Dalam beberapa kasus, AWS Ground Station mungkin mengalami kegagalan yang mencegah data dikirim ke akun Anda, tetapi tidak mengakibatkan kontak berada dalam AWS_CANCELLEDstatus AWS_FAILEDatau. Ketika ini terjadi, AWS Ground Station sebaiknya posting acara khusus akun ke dasbor AWS Kesehatan Anda. Untuk informasi selengkapnya tentang dasbor AWS Kesehatan, lihat Panduan Pengguna AWS Kesehatan.

Memecahkan masalah telemetri

Gunakan informasi berikut untuk memecahkan masalah umum dengan telemetri.

Masalah penyiapan umum

Kesalahan izin IAM

Gejala

Saat menelepon `CreateConfig` untuk membuat TelemetrySinkConfig, Anda menerima kesalahan:

```
Unable to write to Kinesis Data Streams stream. Ensure that Ground Station has
kinesis:PutRecord permissions for the given stream
```

Penyebab

- Peran IAM yang ditentukan dalam TelemetrySinkConfig tidak memiliki izin yang diperlukan untuk menulis ke aliran Kinesis Data Streams.
- Kebijakan kepercayaan pada peran IAM tidak memungkinkan AWS Ground Station untuk mengambil peran tersebut.
- TelemetrySinkConfig Arn aliran Data Kinesis Data Streams di salah atau aliran tidak ada.

Solusi

1. Verifikasi peran IAM ada dan memiliki izin yang benar. Tinjau Langkah 2: Buat TelemetrySinkConfig dan pastikan semua langkah diikuti.

2. Periksa yang AWS Ground Station dapat mengambil peran IAM Anda:

```
aws iam get-role --role-name GroundStationTelemetryRole
```

   Verifikasi kebijakan kepercayaan termasuk `groundstation.amazonaws.com` sebagai prinsip layanan tepercaya.

3. Verifikasi peran IAM memiliki izin Kinesis yang diperlukan:

```
aws iam list-attached-role-policies --role-name GroundStationTelemetryRole
```

   Pastikan kebijakan tersebut mencakup`kinesis:DescribeStream`,`kinesis:PutRecord`, dan `kinesis:PutRecords` izin untuk streaming Anda.

4. Verifikasi aliran Kinesis Data Streams ada dan ARN sudah benar:

```
aws kinesis describe-stream \
    --stream-name your-stream-name \
    --region us-east-2
```

5. Jika menggunakan enkripsi yang dikelola pelanggan, verifikasi peran IAM memiliki `kms:GenerateDataKey` izin untuk kunci Anda.AWS KMS

PassRole kesalahan izin

Gejala

Saat meneleponCreateConfig, Anda menerima kesalahan tentang tidak memiliki izin untuk lulus peran IAM.

Solusi

Pastikan pengguna atau peran IAM Anda memiliki `iam:PassRole` izin untuk peran IAM telemetri. Tambahkan kebijakan berikut ke pengguna atau peran Anda:

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

```
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::99999999999:role/your-stream-name"
    }
  ]
}
```

Masalah konfigurasi aliran Kinesis Data Streams

Gejala

Pengiriman telemetri gagal atau terputus-putus.

Penyebab

- Aliran Kinesis Data Streams memiliki kapasitas yang tidak mencukupi untuk throughput telemetri.
- Aliran sedang digunakan oleh aplikasi lain, menyebabkan pelambatan penulisan.

Solusi

1. Periksa status streaming:

```
aws kinesis describe-stream \
    --stream-name your-stream-name \
    --region us-east-2
```

2. Pantau pelambatan tulis menggunakan CloudWatch metrik:

```
aws cloudwatch get-metric-statistics \
    --namespace AWS/Kinesis \
    --metric-name WriteProvisionedThroughputExceeded \
    --dimensions Name=StreamName,Value=your-stream-name \
    --start-time 2025-12-08T00:00:00Z \
    --end-time 2025-12-08T23:59:59Z \
    --period 60 \
    --statistics Sum \
    --region us-east-2
```

3. Jika pelambatan terdeteksi, pertimbangkan:

    - Beralih ke mode kapasitas sesuai permintaan untuk penskalaan otomatis.

- Menggunakan aliran khusus untuk AWS Ground Station telemetri.
- Jika menggunakan mode yang disediakan, tingkatkan jumlah pecahan.

Masalah pengiriman telemetri

Tidak ada data telemetri yang muncul

Gejala

Setelah menjadwalkan kontak dengan profil misi berkemampuan telemetri, tidak ada data telemetri yang muncul di aliran Kinesis Data Streams Anda.

Kemungkinan penyebab dan solusi

Profil misi tidak mengaktifkan telemetri

Verifikasi profil misi yang digunakan untuk kontak termasuk`telemetrySinkConfigArn`:

```
aws groundstation get-mission-profile \
    --mission-profile-id 12345678-1234-1234-1234-123456789012 \
    --region us-east-2
```

Periksa output untuk `telemetrySinkConfigArn` bidang tersebut. Jika tidak ada, profil misi tidak mengaktifkan telemetri.

Masalah izin peran IAM

Tinjau langkah-langkah pemecahan masalah izin IAM di. Kesalahan izin IAM

Aliran Kinesis Data Streams tidak ada atau berada di wilayah yang salah

Verifikasi aliran ada di wilayah yang benar:

```
aws kinesis describe-stream \
    --stream-name your-stream-name \
    --region us-east-2
```

Kontak belum dimulai

Pengiriman telemetri dimulai pada waktu mulai kontak. Verifikasi kontak telah dimulai dengan memeriksa status kontak:

```
aws groundstation describe-contact \
    --contact-id 12345678-1234-1234-1234-123456789012 \
    --region us-east-2
```

Kontak dijadwalkan di stasiun darat kembar digital

Stasiun ground kembar digital tidak mendukung pengiriman telemetri. Kontak yang dijadwalkan pada antena kembar digital tidak menghasilkan data telemetri. Verifikasi bahwa stasiun bumi yang digunakan untuk kontak Anda bukan kembar digital dengan memeriksa nama stasiun bumi. Stasiun ground kembar digital memiliki awalan “Digital Twin” atas nama mereka. Untuk informasi selengkapnya, lihat Gunakan AWS Ground Station fitur kembar digital.

Data telemetri intermiten

Gejala

Data telemetri disampaikan secara tidak konsisten dengan kesenjangan atau catatan yang hilang.

Kemungkinan penyebabnya

- Kinesis Data Streams mengalirkan masalah kapasitas atau pelambatan. Lihat Masalah konfigurasi aliran Kinesis Data Streams.
- Masalah konektivitas jaringan antara AWS Ground Station dan aliran Kinesis Data Streams Anda.

Solusi

- Pantau CloudWatch metrik aliran Data Streams Kinesis untuk pelambatan atau kesalahan.
- Pastikan streaming Anda menggunakan mode kapasitas sesuai permintaan atau memiliki kapasitas yang cukup.
- Gunakan aliran khusus untuk AWS Ground Station telemetri untuk menghindari pertengkaran dengan aplikasi lain.

Masalah format data

Kesalahan penguraian JSON

Gejala

Aplikasi Anda mengalami kesalahan saat mengurai catatan telemetri sebagai JSON.

Solusi

- Verifikasi decoding Base64 - Data dalam aliran Kinesis Data Streams adalah. Base64-encoded Pastikan Anda memecahkan kode data sebelum menguraikannya sebagai JSON. Untuk informasi selengkapnya, lihat Membaca data dari aliran Kinesis Data Streams.
- Periksa catatan kosong -AWS Ground Station dapat mengirim catatan validasi kosong saat membuat TelemetrySinkConfigfile. Aplikasi Anda harus menangani catatan kosong atau cacat dengan anggun.
- Terapkan parsing sadar versi - Parse the`telemetryTypeAndVersion`,`telemetryType`, and `telemetryVersion` fields terlebih dahulu untuk menentukan skema yang sesuai untuk setiap record.

Jenis atau versi telemetri yang tidak diketahui

Gejala

Aplikasi Anda menemukan jenis atau versi telemetri yang tidak dikenalinya.

Solusi

Ini adalah perilaku yang diharapkan karena jenis telemetri baru dan versi skema dapat diperkenalkan dari waktu ke waktu. Aplikasi Anda harus:

- Log jenis dan versi yang tidak dikenal untuk pemantauan.
- Lanjutkan memproses jenis dan versi yang dikenal.
- Menerapkan penanganan yang anggun untuk skema yang tidak diketahui.

Untuk informasi selengkapnya tentang pembuatan versi skema, lihat. Pembuatan versi dan evolusi skema

Mendapatkan bantuan

Jika Anda terus mengalami masalah setelah mengikuti langkah-langkah pemecahan masalah, hubungi Support AWS.

Informasi untuk diberikan

Saat menghubungi dukungan, berikan informasi berikut:

- ID kontak yang mengalami masalah
- ID profil misi yang digunakan
- TelemetrySinkConfig ARN
- Aliran Data Streams Kinesis ARN
- IAM berperan ARN dan kebijakan terlampir
- Pesan galat dari CloudWatch Log atau aplikasi Anda
- Stempel waktu saat masalah terjadi
- Langkah pemecahan masalah sudah diambil

Untuk AWS Ground Station dukungan umum, lihat Panduan AWS Ground Station Pengguna.

Kuota dan batas

Anda dapat melihat wilayah yang didukung, titik akhir terkait, dan kuota di AWS Ground Station titik akhir dan kuota.

Anda dapat menggunakan konsol Kuota Layanan, AWS API, dan AWS CLI untuk meminta peningkatan kuota, bila diperlukan.

Ketentuan layanan

Untuk persyaratan AWS Ground Station layanan, silakan merujuk ke Ketentuan Layanan AWS.

Sejarah Dokumen untuk AWS Ground Station Panduan Pengguna

Tabel berikut menjelaskan perubahan penting dalam setiap rilis Panduan AWS Ground Station Pengguna.

Perubahan	Deskripsi	Tanggal
Fitur Baru	Menambahkan dokumentasi untuk dukungan nomor satelit Alpha-5 TLE.AWS Ground Station sekarang menerima ephemerides TLE dengan nomor satelit yang Alpha-5 dikodekan untuk nomor katalog satelit dalam kisaran 100.000—339.999. Untuk informasi lebih lanjut, lihat Menyediakan data ephemeris TLE.	28 Mei 2026
Fitur Baru	Ditambahkan dokumentasi untuk Antena AWS Ground Station Khusus. Untuk informasi lebih lanjut, lihat Antena AWS Ground Station Khusus.	April 14, 2026
Fitur Baru	Menambahkan dokumenta si untuk UpdateContact API dan versi kontak. Untuk informasi selengkapnya, lihat Memperbarui kontak dan pembuatan versi kontak.	April 14, 2026

Fitur Baru	Menambahkan dokumentasi untuk ListAntennas dan ListGroundStationReservations API. Untuk informasi lebih lanjut, lihat AWS Ground Station Lokasi dan Lihat reservasi stasiun bumi.	April 14, 2026
Pembaruan Dokumentasi	Menambahkan fungsionalitas tambahan ke CancelContact API, dan menyertakan informasi tentang fungsionalitas tersebut dan implikasi pengukuran. Untuk informasi selengkapnya lihat Memahami pengukuran kontak.	Desember 10, 2025
Pembaruan Dokumentasi	Diklarifikasi bahwa CloudWatch metrik dipancarkan di wilayah yang terkait dengan stasiun bumi kontak. Memperbaiki tautan yang rusak.	Desember 2, 2025
Kebijakan AWS terkelola yang diperbarui	AWS Ground Station telah memperbarui kebijakan terkelola `AWSGroundStationAgentInstancePolicy` untuk menyertakan izin tambahan untuk mengambil URL respons tugas. Untuk selengkapnya, lihat AWS Ground Station pembaruan kebijakan AWS terkelola.	November 13, 2025

Fitur Baru	Memperbarui panduan pengguna untuk memasukkan ephemerides elevasi azimuth. Untuk informasi lebih lanjut, lihat Menyediakan data ephemeris elevasi azimuth	Oktober 22, 2025
Pembaruan Dokumentasi	Cross-region pengiriman data tidak lagi memerluka n konfigurasi atau persetuju an khusus. Untuk informasi selengkapnya, lihat Menggunakan pengiriman data lintas wilayah.	September 11, 2025
Pembaruan Dokumentasi	Menambahkan klarifikasi tentang pemanfaatan kontak sumber daya yang dikonfigu rasi.	April 4, 2025
Fitur Baru	Memperbarui panduan pengguna untuk menyertakan kembar AWS Ground Station digital.	Agustus 6, 2024
Pembaruan Dokumentasi	Memperbarui banyak bagian dari panduan pengguna, termasuk diagram baru, contoh, dan banyak lagi.	Juli 18, 2024
Pembaruan Dokumentasi	Menambahkan umpan RSS ke Panduan Pengguna.	Juli 18, 2024
Pembaruan Dokumentasi	Pisahkan Panduan Pengguna AWS Ground Station Agen menjadi Panduan Pengguna terpisah.	Juli 18, 2024

Fitur Baru	Kontak sekarang dapat dijadwalkan hingga 30 detik di luar rentang waktu visibilit as. Waktu visibilitas termasuk dalam DescribeContact tanggapan.	26 Maret 2024
Pembaruan Dokumentasi	Organisasi yang ditingkatkan dan menambahkan bagian “Pemilihan Instans EC2 dan Perencanaan CPU”.	Maret 6, 2024
Pembaruan Dokumentasi	Menambahkan praktik terbaik baru ke Panduan Pengguna AWS Ground Station Agen untuk menjalankan layanan dan proses bersama AWS Ground Station Agen.	Februari 23, 2024
Pembaruan Dokumentasi	Ditambahkan halaman Catatan Rilis Agen.	Februari 21, 2024
Pembaruan Template	Ditambahkan dukungan untuk subnet publik terpisah dalam DirectBroadcastSatelliteWbD igIfEc2DataDelivery template.	Februari 14, 2024
Pembaruan Dokumentasi	Menambahkan rujukan ke AWS Notifikasi Pengguna dalam dokumentasi pemantauan.	Agustus 6, 2023
Pembaruan Dokumentasi	Menambahkan instruksi untuk menandai satelit dengan nama yang akan ditampilkan di konsol.AWS Ground Station	26 Juli 2023

Fitur Baru	Menambahkan Panduan Pengguna AWS Ground Station Agen untuk rilis Pengiriman Data DiGIF Wideband.	12 April 2023
Kebijakan AWS terkelola baru	AWS Ground Station menambahkan kebijakan baru bernama AWSGround StationAgentInstancePolicy.	12 April 2023
Fitur Baru	Memperbarui panduan pengguna untuk rilis Pratinjau CPE.	9 November 2022
Kebijakan AWS terkelola baru	AWS Ground Station menambahkan AWSServic eRoleForGroundStationDataflowEndpointGroup service-linked-role (SLR) yang menyertakan kebijakan baru bernama. AWSServic eRoleForGroundStationDataflowEndpointGroupPolicy	2 November 2022
Fitur Baru	Memperbarui panduan pengguna untuk menyertakan integrasi dengan AWS CLI.	17 April 2020
Fitur Baru	Memperbarui panduan pengguna untuk menyertakan integrasi dengan CloudWatch Metrik.	24 Februari 2020

Template Baru	Satelit Siaran Publik (AquaSnppJpss Template) ditambahkan ke Panduan Pengguna.AWS Ground Station	19 Februari 2020
Fitur Baru	Memperbarui panduan pengguna untuk menyertakan pengiriman data lintas wilayah.	5 Februari 2020
Pembaruan Dokumentasi	Contoh dan deskripsi yang diperbarui untuk pemantauan AWS Ground Station dengan CloudWatch Acara.	4 Februari 2020
Pembaruan Dokumentasi	Lokasi template telah diperbarui dan bagian Memulai dan Pemecahan Masalah telah direvisi.	19 Desember 2019
Bagian Pemecahan Masalah Baru	Bagian pemecahan masalah ditambahkan ke AWS Ground Station Panduan Pengguna.	7 November 2019
Topik Memulai Baru	Memperbarui topik Memulai, yang mencakup CloudForm ation template terbaru.	1 Juli 2019
Versi Kindle	Versi Kindle yang diterbitkan dari Panduan AWS Ground Station Pengguna.	20 Juni 2019
Layanan dan panduan baru	Ini adalah rilis awal AWS Ground Station dan Panduan AWS Ground Station Pengguna.	23 Mei 2019

AWS Glosarium

Untuk AWS terminologi terbaru, lihat AWS glosarium di Referensi.Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.