aws

ユーザーガイド

AWS Ground Station

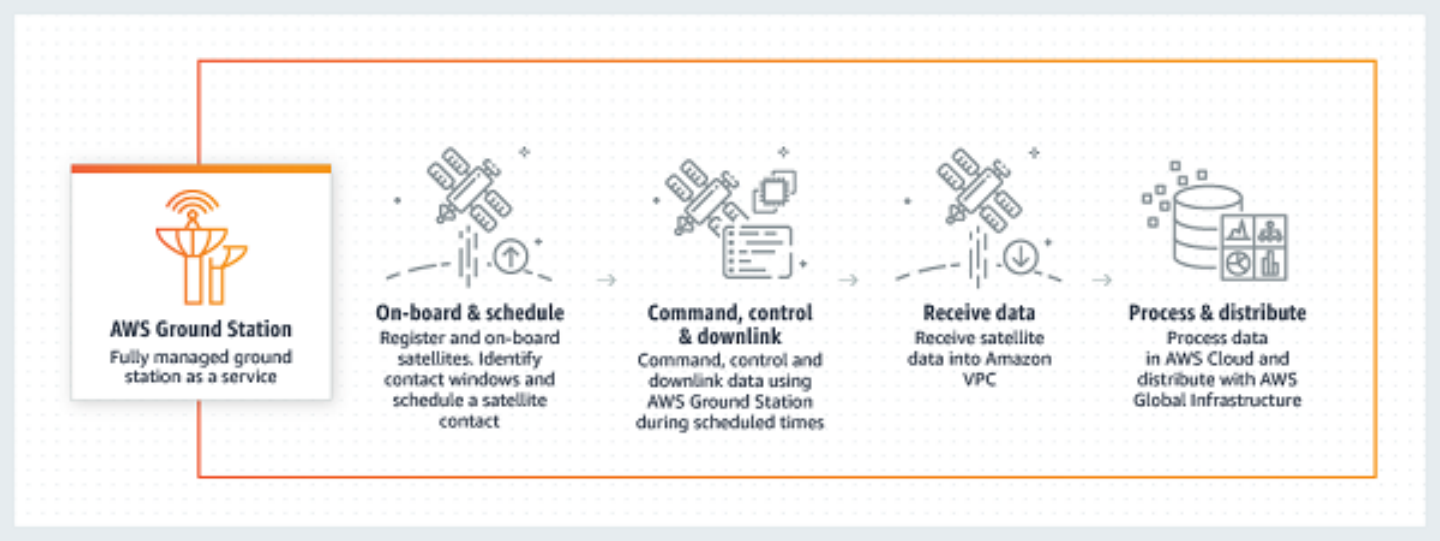
AWS Ground Station: ユーザーガイド

Table of Contents

AWS Ground Station とは?

AWS Ground Station は、衛星との通信の制御、衛星データの処理、および衛星の運用のスケールを行うことができるフルマネージド型のサービスです。つまり、今後は自分のGround Stationのインスラストラクチャを構築したり管理したりする必要はありません。

AWS Ground Station を使用すれば、独自の地上局の運用および保守にリソースを費やすよりも、衛星データを取り込みサーバーやストレージの使用を動的にスケールする新しいアプリケーションの発明や迅速な実験に注力できます。



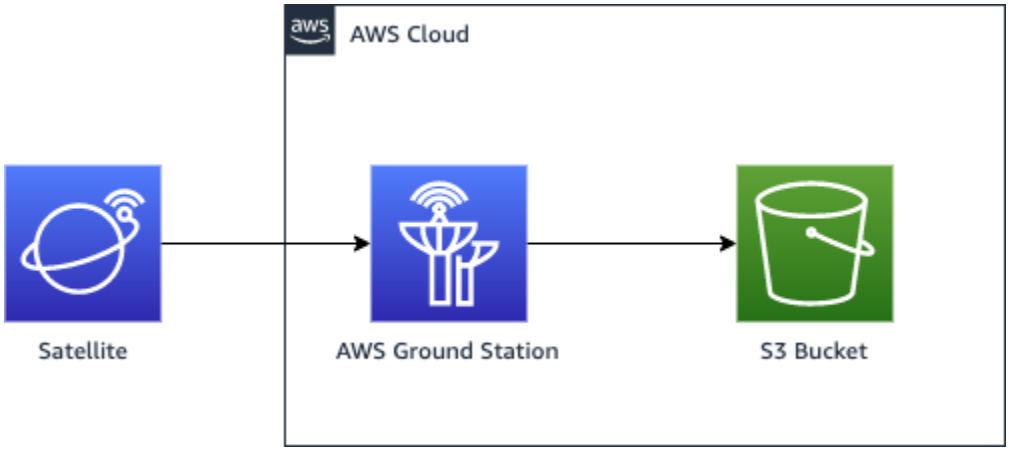
AWS Ground Station の仕組み

衛星の予約はコンタクトと呼ばれます。衛星はコンタクト中に AWS Ground Station アンテナと通信します。場所、時間、ミッション情報を指定することで、API または AWS コンソールを介して連絡先を予約できます。コンタクトデータは、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスに対してストリーミングすることや、アカウント内の Amazon Simple Storage Service (Amazon S3) バケットに非同期で配信することができます。

拡張可能で再利用可能な設定リソースを作成して、コンタクト中の AWS Ground Station アンテナの設定方法を制御できます。 ミッションプロファイルを使用して、データの取得場所、使用すべきフォーマット、送信先を指定できます。

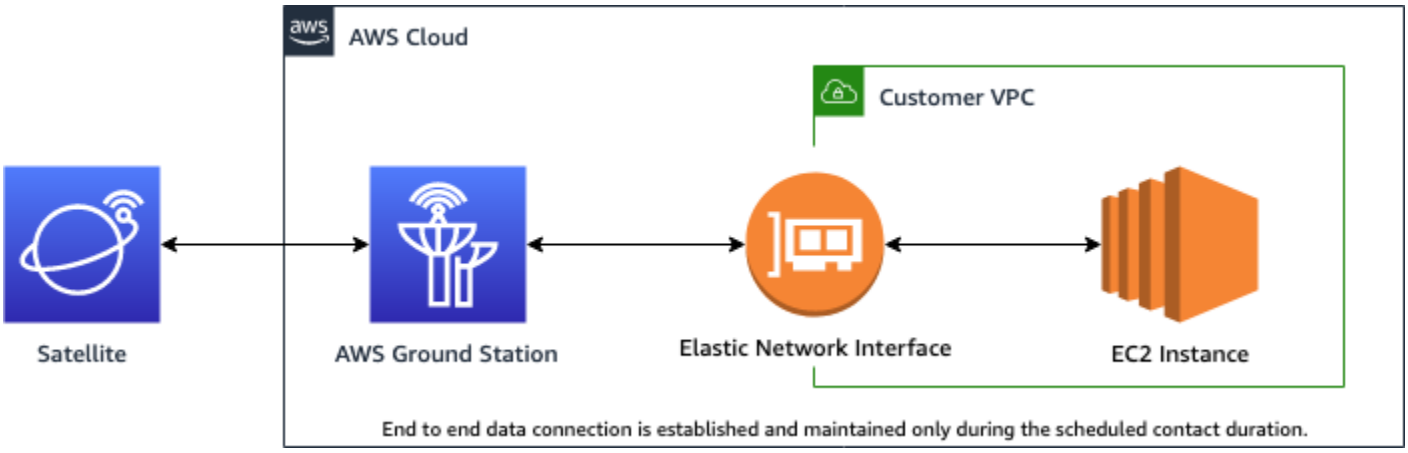
Amazon S3 へのデータ配信

Amazon S3 へのデータ配信では、アカウント内の Amazon S3 バケットにコンタクトデータが非同期に配信されます。コンタクトデータは、コンタクトデータをソフトウェア定義無線 (SDR) に再生するため、または処理を目的としてパケットキャプチャ (pcap) ファイルからペイロードデータを抽出するために pcap ファイルとして配信されます。pcap ファイルは、コンタクトデータがアンテナハードウェアによって受信されると、30 秒ごとに Amazon S3 バケットに配信され、必要に応じてコンタクト中にコンタクトデータを処理できます。受信したら、独自の後処理ソフトウェアを使用してデータを処理したり、Amazon SageMaker や Amazon Rekognition などの他の AWS のサービスを使用したりできます。Amazon S3 へのデータ配信は、衛星からのデータのダウンリンクにのみ使用できます。Amazon S3 から衛星にデータをアップリンクすることはできません。



Amazon EC2 へのデータ配信

Amazon EC2 へのデータ配信では、コンタクトデータは Amazon EC2 インスタンスに対してストリーミングされます。Amazon EC2 インスタンスでデータをリアルタイムで処理することや、後処理のためにデータを転送することができます。



詳細情報

AWS Ground Station を使用すると、衛星通信を介して 125 を超えるサービスにアクセスできます。次の点に注意してください。

- ナローバンド RF データは、最大 54 MHz の帯域幅で S バンド (2200 ~ 2300 MHz) または X バンド (7750 ~ 8400 MHz) で受信できます。
  - S バンド RF データはデジタル化され、VITA-49 信号データ/IP 形式のデジタルストリームとして提供されます。
  - X バンド中間周波数 (IF) データは、デジタル化され、VITA-49 信号データ/IP 形式のデジタルストリームとして提供されます。
- ブロードバンドの復調/デコードされたデータは、最大 500 MHz の帯域幅で X バンド (7750 ~ 8400 MHz) で受信できます。
  - X バンド中間周波数 (IF) データは、復調およびデコードされ、VITA-49 拡張データ/IP 形式のデジタルストリームとして提供されます。
- AWS Ground Station エージェントを介して、40 MHz から 400 MHz の帯域幅まで広帯域デジタル中間周波数 (DigIF) データを受信できます。
  - AWS Ground Station エージェントおよび広帯域 DigIF データ配信の詳細については、AWS Ground Station エージェントユーザーガイド「」を参照してください。
- RF データは、最大 54 MHz の帯域幅で S バンド (2025〜2120 MHz) で送信できます。

  - RF データは、VITA-49 信号データ/IP 形式のデジタルストリーム AWS Ground Station としてに提供されます。
- をサポートする AWS リージョン AWS Ground Station から を実行する必要があります AWS Ground Station。サポートされているリージョンのリストについては、グローバルインフラストラクチャの「リージョン表」を参照してください。
- アンテナと同じリージョンで実行されている Amazon EC2 インスタンスにデータを配信したり、クロスリージョンデータ配信を使用して、アンテナから任意の AWS リージョンの Amazon EC2 インスタンスにデータを送信したりできます。現在、以下の antenna-to-destination リージョンが利用可能です。
  - 米国東部 (オハイオ) リージョン (us-east-2) から米国西部 (オレゴン) リージョン (us-west-2)
  - 米国西部 (オレゴン) リージョン (us-west-2) から米国東部 (オハイオ) リージョン (us-east-2)

サービス条件

本サービスは、お客様が所有する、または使用許諾を受けた、または法的に取得したコンテンツの保存、取得、照会、提供、および実行のためにのみ使用できます。これらのサービス条件で使用されている (a)「サービス利用者コンテンツ」には「企業コンテンツ」と「カスタマーコンテンツ」が含まれ、(b)「AWS コンテンツ」には「Amazon のプロパティ」が含まれます。サービスの一部として、当社またはサードパーティーのライセンサーによって提供される特定のソフトウェア (関連ドキュメントを含む) の使用が許可されることがあります。

> **Important**
>
> このソフトウェアは販売または配布されるものではなく、本サービスの一部としてのみ使用することができます。特定の許可なくして、本サービスの外部に転送することはできません。

コアコンポーネント

データフローエンドポイントグループ、設定、ミッションプロファイルは、 のコアコンポーネントです AWS Ground Station。これらのコンポーネントは、コンタクトのスケジュール方法、アンテナと人工衛星の通信方法、およびデータの配信先を決定します。の使用を開始する前に AWS Ground Station、これらのコンポーネントについて学習することをお勧めします。例については、それぞれのセクションで示します。

トピック

データフローエンドポイントグループ

データフローエンドポイントは、コンタクト中のデータストリームの受け渡し場所を定義します。エンドポイントは、コンタクトの実行時に選択した名前によって識別されます。これらの名前は一意である必要はありません。これにより、同じミッションプロファイルを使用して同時に複数のコンタクトを実行できます。

エンドポイントリストのアドレスには次が含まれます。

- `name` - このデータフローのエンドポイントの IP アドレス。
- `port` - 接続先のポート。

エンドポイントのセキュリティ詳細は、次が含まれます。

- `roleArn` - VPC に Elastic Network Interface (ENIs) を作成するために AWS Ground Station が引き受けるロールの Amazon リソースネーム (ARN)。これらの ENI は、コンタクト中にストリーミングされるデータの入出力ポイントとなります。
- `securityGroupIds` -Elastic Network Interface にアタッチするセキュリティグループ。
- `subnetIds` - がインスタンスにストリームを送信する Elastic Network Interface AWS Ground Station を配置するサブネットのリスト。

`roleArn` に渡される IAM ロールには、`groundstation.amazonaws.com` サービスプリンシパルがロールを引き受けることを許可する信頼ポリシーが必要です。例については、以下の「信頼ポリシーの例」を参照してください。エンドポイントの作成時にエンドポイントリソース ID が存在しないため、信頼ポリシーは の代わりにアスタリスク (*) を使用する必要があります*your-endpoint-id*。作成後にエンドポイントリソース ID を使用してこれを更新し、信頼ポリシーをその特定のデータフローエンドポイントグループに絞り込むことができます。

IAM ロールには、 が ENIs AWS Ground Station をセットアップできるようにする IAM ポリシーが必要です。例については、以下の「ロールポリシーの例」を参照してください。

信頼ポリシーの例

ロールの信頼ポリシーを更新する方法の詳細については、IAM ユーザーガイドの「IAM ロールの管理」を参照してください。

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

ロールポリシーの例

ロールポリシーを更新またはアタッチする方法の詳細については、IAM ユーザーガイドの「IAM ポリシーを管理する」を参照してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
```

```
        "ec2:CreateNetworkInterfacePermission",
        "ec2:DeleteNetworkInterfacePermission",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs",
        "ec2:DescribeSecurityGroups"
      ]
    }
  ]
}
```

データフローエンドポイントは、常にデータフローエンドポイントグループの一部として作成されます。1 つのグループに複数のデータフローエンドポイントを含めることで、1 回のコンタクトで指定されたエンドポイントをすべて一緒に使用できることを断定できます。たとえば、コンタクトが 3 つの別々のデータフローエンドポイントにデータを送信する必要がある場合、1 つのデータフローエンドポイントグループに、ミッションプロファイルのデータフローエンドポイント設定と一致するエンドポイントが 3 つ必要です。

データフローエンドポイントグループ内の 1 つ以上のリソースがコンタクトに使用されている場合、グループ全体がそのコンタクトの間リザーブされます。複数のコンタクトを同時に実行できますが、それらのコンタクトは異なるデータフローエンドポイントグループで実行する必要があります。

データフローエンドポイントグループは、それらを使用するコンタクトをスケジュールするために HEALTHY 状態になっている必要があります。データフローエンドポイントグループが HEALTHY の状態にならない可能性がある理由と、取るべき適切な是正措置を以下に示します。

- NO_REGISTERED_AGENT - EC2 インスタンスを起動します。これにより、エージェントが登録されます。この呼び出しが成功するには、有効なコントローラー設定ファイルが必要であることに注意してください。このファイルの設定の詳細については、「AWS Ground Station エージェントユーザーガイド」を参照してください。
- INVALID_IP_OWNERSHIP - DeleteDataflowEndpointGroup API を使用してデータフローエンドポイントグループを削除し、次に CreateDataflowEndpointGroup API を使用して EC2 インスタンスに関連付けられている IP アドレスとポートを使用してデータフローエンドポイントグループを再作成します。
- UNVERIFIED_IP_OWNERSHIP - IP アドレスはまだ検証されていません。検証は定期的に行われるため、これは、自動的に解決するはずです。
- NOT_AUTHORIZED_TO_CREATE_SLR - アカウントに、必要なサービスにリンクされたロールを作成する権限がありません。「Ground Station のサービスにリンクされたロールの使用」のトラブルシューティングの手順を確認してください。

、 AWS CloudFormation、または AWS Ground Station API を使用してデータフローエンドポイントグループに対してオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::DataflowEndpointグループ CloudFormation リソースタイプ
- データフローエンドポイントグループの AWS CLI リファレンス
- cCreateDataflowEndpointGroup API リファレンス

設定

Config は、 が問い合わせの各側面のパラメータを定義 AWS Ground Station するために使用するリソースです。希望する設定をミッションプロファイルに追加すると、コンタクトを実行する際にそのミッションプロファイルが使用されます。さまざまなタイプの設定を定義できます。

AWS CloudFormation、、または AWS Ground Station API を使用して設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。特定の設定タイプのドキュメントへのリンクも以下に記載されています。

- AWS::GroundStation::Config CloudFormation リソースタイプ
- Config AWS CLI リファレンス
- CreateConfig API リファレンス

データフローエンドポイント設定

> **Note**
>
> データフローエンドポイント設定は、Amazon EC2 へのデータ配信にのみ使用され、Amazon S3 へのデータ配信には使用されません。

データフローエンドポイント設定を使用して、コンタクト中にデータのフローを行うデータフローエンドポイントグループ内のデータフローエンドポイントを指定します。データフローエンドポイント設定の 2 つのパラメータは、データフローエンドポイントの名前とリージョンを指定します。コンタクトを予約すると、 は指定したミッションプロファイル AWS Ground Station を分析し、ミッションプロファイルに含まれるデータフローエンドポイント設定で指定されたすべてのデータフローエンドポイントを含むデータフローエンドポイントグループを見つけようとします。

データフローエンドポイント設定の dataflowEndpointName プロパティは、コンタクト中にデータのフローを行うデータフローエンドポイントグループ内のデータフローエンドポイントを指定します。

dataflowEndpointRegion プロパティは、データフローエンドポイントが存在するリージョンを指定します。データフローエンドポイント設定でリージョンが指定されている場合、 は指定されたリージョン内のデータフローエンドポイント AWS Ground Station を検索します。リージョンが指定されていない場合、 AWS Ground Station デフォルトでコンタクトのグラウンドステーションリージョンになります。データフローエンドポイントのリージョンがコンタクトの地上ステーションリージョンと同じでない場合、コンタクトはクロスリージョンデータ配信コンタクトとみなされます。

、 AWS CloudFormation、または AWS Ground Station API を使用してデータフローエンドポイント設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config DataflowEndpointConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (dataflowEndpointConfig -> (structure)「」セクションを参照)
- DataflowEndpointConfig API リファレンス

S3 記録設定

> **Note**
>
> S3 記録設定は、Amazon S3 へのデータ配信にのみ使用され、Amazon EC2 へのデータ配信には使用されません。

S3 記録設定を使用して、ダウンリンクされたデータの配信先 Amazon S3 バケットを指定できます。S3 記録設定の 2 つのパラメータは、Amazon S3 バケットにデータを配信するときに が引き受け AWS Ground Station る Amazon S3 バケットと IAM ロールを指定します。指定された IAM ロールと Amazon S3 バケットは、以下の条件を満たす必要があります。

- Amazon S3 バケットの名前は、aws-groundstation で始まる必要があります。
- IAM ロールには、groundstation.amazonaws.com サービスプリンシパルがロールを引き受けることを許可する信頼ポリシーが必要です。例については、以下の「信頼ポリシーの例」を参照してください。設定の作成時に設定リソース ID が存在しない場合、信頼ポリシーは の代わりにアス

タリスク (*) を使用する必要がありyour-config-id、設定リソース ID を使用して作成後に更新できます。

- IAM ロールには、バケット上で s3:GetBucketLocation アクションとバケットのオブジェクト上での s3:PutObject の実行を許可する IAM ポリシーが必要です。Amazon S3 バケットにバケットポリシーがある場合、バケットポリシーは IAM ロールでこれらのアクションの実行を許可する必要があります。例については、以下の「ロールポリシーの例」を参照してください。

信頼ポリシーの例

ロールの信頼ポリシーを更新する方法の詳細については、IAM ユーザーガイドの「IAM ロールの管理」を参照してください。

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

ロールポリシーの例

ロールポリシーを更新またはアタッチする方法の詳細については、IAM ユーザーガイドの「IAM ポリシーを管理する」を参照してください。

```
{
  "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/*"
      ]
    }
  ]
}
```

、、または AWS Ground Station API を使用して S3 記録設定でオペレーションを実行する方法の詳細については AWS CloudFormation AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config S3RecordingConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (`s3RecordingConfig -> (structure)`「」セクションを参照)
- S3RecordingConfig API リファレンス

追跡設定

ミッションプロファイルの追跡設定を使用して、コンタクト中に自動追跡を有効にする必要があるかどうかを決定できます。この設定には単一のパラメータ、`autotrack` があります。この `autotrack` パラメータには以下の値があります。

- REQUIRED -コンタクトに自動追跡が必要。
- PREFERRED -コンタクトに自動追跡が好ましいが、自動追跡がなくてもコンタクトを実行できる。
- REMOVED - コンタクトに自動追跡が使用されるべきではない。

AWS CloudFormation、、または AWS Ground Station API を使用して追跡設定に対してオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config TrackingConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (`trackingConfig -> (structure)`「」セクションを参照)
- TrackingConfig API リファレンス

アンテナダウンリンク設定

アンテナダウンリンク設定を使用してコンタクト中のダウンリンク用のアンテナを設定できます。これらは、ダウンリンクコンタクト中に使用すべき周波数、帯域幅、および偏波を指定するスペクトル設定で構成されています。ダウンリンクのユースケースで復調や復号が必要な場合は、「アンテナダウンリンク復調デコード設定」を参照してください。

、 AWS CloudFormation、または AWS Ground Station API を使用してアンテナダウンリンク設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config AntennaDownlinkConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (`antennaDownlinkConfig -> (structure)`「」セクションを参照)
- AntennaDownlinkConfig API リファレンス

アンテナダウンリンク復調デコード設定

アンテナダウンリンク復調デコード設定はより複雑でカスタマイズ可能な設定タイプです。これを使用して、復調またはデコードでダウンリンクコンタクトを実行できます。これらのタイプのコンタクトを実行することに関心がある場合は、 AWS Ground Station チームにお問い合わせください。ユースケースに適した設定とミッションプロファイルを定義するお手伝いをします。

、 AWS CloudFormation、または AWS Ground Station API を使用してアンテナダウンリンク復調デコード設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config AntennaDownlinkDemodDecodeConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (`antennaDownlinkDemodDecodeConfig -> (structure)`「」セクションを参照)
- AntennaDownlinkDemodDecodeConfig API リファレンス

アンテナアップリンク設定

アンテナアップリンク設定を使用してコンタクト中のアップリンクのアンテナを設定できます。これらは、周波数、偏波、および目標実効等方輻射電力 (EIRP) を含むスペクトル設定で構成されています。アップリンクループバックのコンタクトを設定する方法については、「アップリンクエコー設定」を参照してください。

、 AWS CloudFormation、または AWS Ground Station API を使用してアンテナアップリンク設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config AntennaUplinkConfig CloudFormation プロパティ
- Config AWS CLI リファレンス (`antennaUplinkConfig -> (structure)`「」セクションを参照)
- AntennaUplinkConfig API リファレンス

アップリンクエコー設定

アップリンクエコー設定は、アップリンクエコーを実行する方法をアンテナに伝えます。これにより、アンテナから送信された信号をデータフローエンドポイントにそのまま送り返します。アップリンクエコー設定には、アップリンク設定の ARN が含まれています。アンテナは、アップリンクエコーを実行する際に ARN により指定されたアップリンク設定からのパラメータを使用します。

、 AWS CloudFormation、または AWS Ground Station API を使用してアップリンクエコー設定でオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::Config UplinkEchoConfig CloudFormation プロパティ

- Config AWS CLI リファレンス (`uplinkEchoConfig -> (structure)`「」セクションを参照)
- UplinkEchoConfig API リファレンス

ミッションプロファイル

ミッションプロファイルには、コンタクトの実行方法に関する設定とパラメータが含まれています。コンタクトを予約したり利用可能なコンタクトを検索したりするときには、使用する予定のミッションプロファイルを指定します。ミッションプロファイルはすべての設定を1つにまとめ、コンタクト中のアンテナの設定方法とデータの配信先を定義します。

追跡設定に加え、すべての設定がミッションプロファイルの `dataflowEdges` フィールドに含まれます。単一のデータフローのエッジは 2 つの ARN のリストになります。1 番目は config から、2 番目は config 宛てです。2 つの設定間でデータフローエッジを指定することで、コンタクト中にデータがどこに AWS Ground Station 、どこで流れるかを に指示できます。追跡設定はデータフローエッジの一部としては使用されませんが、別のフィールドとして指定されます。

ミッションプロファイルの `name` フィールドを確認すれば、作成したミッションプロファイルと区別できます。

AWS CloudFormation、、または AWS Ground Station API を使用してミッションプロファイルでオペレーションを実行する方法の詳細については AWS Command Line Interface、次のドキュメントを参照してください。

- AWS::GroundStation::MissionProfile CloudFormation リソースタイプ
- ミッションプロファイル AWS CLI リファレンス
- CreateMissionProfile API リファレンス

AWS Ground Station のロケーション

お客様は、米国 (オレゴン)、米国 (オハイオ)、米国 (アラスカ)、中東 (バーレーン)、欧州 (ストックホルム)、アジアパシフィック (ダッボー)、欧州 (アイルランド)、アフリカ (ケープタウン)、米国 (ハワイ)、アジアパシフィック (ソウル)、アジアパシフィック (シンガポール)および南米 (プンタアレナス) の AWS Ground Station アンテナを使用してデータを送受信できます

お客様は、米国西部 (オレゴン)、米国東部 (オハイオ)、中東 (バーレーン)、欧州 (ストックホルム)、アジアパシフィック (ダッボー)、欧州 (アイルランド)、アフリカ (ケープタウン)、米国東部 (バージニア北部)、欧州 (フランクフルト)、アジアパシフィック (ソウル)、アジアパシフィック (シンガポール)および南米 (サンパウロ) の AWS Ground Station コンソールでデータ配信およびコンタクトの設定を行うことができます。

注意: AWS Ground Station リソースは、前の段落で説明した AWS Ground Station コンソールをホストするリージョンでのみ作成できます。

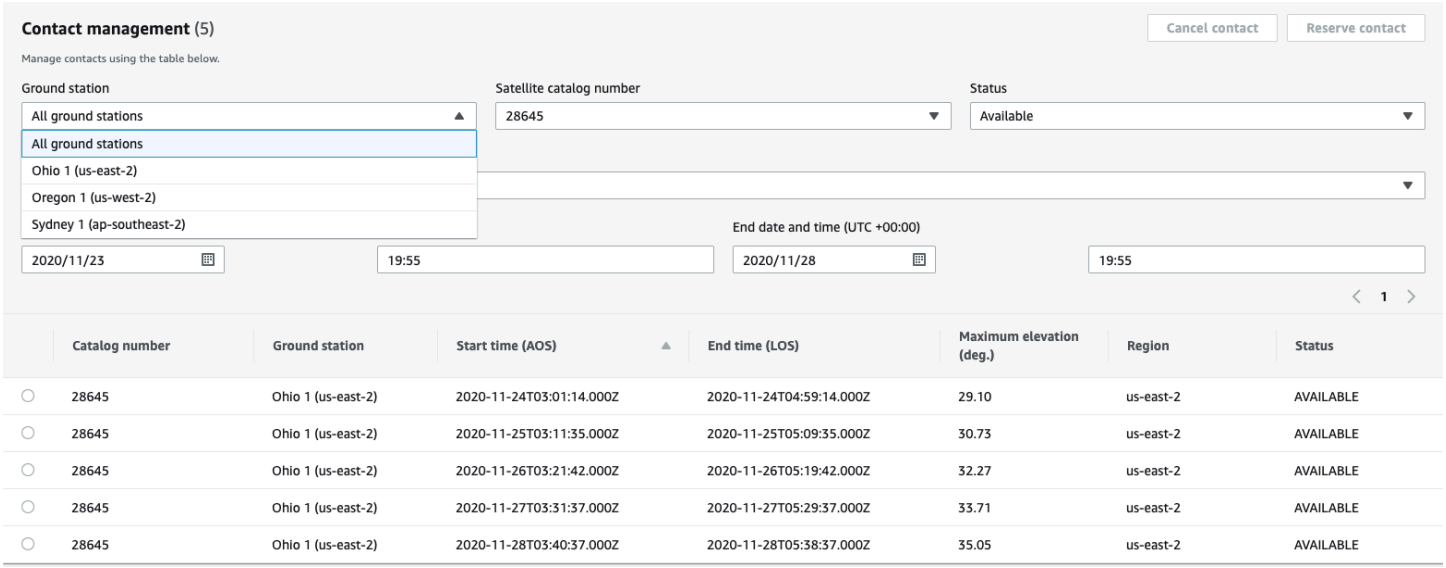
トピック

- Ground Station の AWS リージョンの検索

Ground Station の AWS リージョンの検索

AWS グローバルネットワークには、接続されている AWS リージョンに物理的に配置されていない Ground Station の口ケーションが含まれます。これらの Ground Station 口ケーションでコンタクトの一覧表示と予約は、Ground Station が接続されている AWS リージョンを使用して行う必要があります。

Ground Station の AWS リージョンは複数の方法で決定できます。AWS Ground Station コンソールページには、次の図に示すように、フィルターと連絡先テーブルの両方に Ground Station の AWS リージョンが表示されます。AWS SDK では、ListGroundStation レスポンスに Ground Station の AWS リージョンが含まれます。最後に、AWS CLI では、list-ground-stations レスポンスに Ground Station の AWS リージョンが含まれます。



トピック

- AWS リージョンの外に配置されている Ground Station の例

AWS リージョンの外に配置されている Ground Station の例

Hawaii 1 は、接続先の AWS リージョンに物理的に配置されていないGround Station 口ケーションの例です。Hawaii 1 Ground Station は米国ハワイに配置されていますが、us-west-2 (オレゴン) の AWS リージョンに接続しています。Hawaii 1を使用してコンタクトを一覧表示および予約するには、us-west-2 (オレゴン) の AWS リージョンで設定されたミッションプロファイルが必要で、AWS Ground Station コンソール、AWS CLI、または AWS SDK で us-west-2 (オレゴン) の AWS リージョンを使用する必要があります。

- AWS Ground Station コンソールで Hawaii 1 の予約コンタクトを一覧表示するには、us-west-2 (Oregon) リージョンで AWS Ground Station コンソールを使用する必要があります。
- AWS CLI を使用して Hawaii 1 のコンタクトを一覧表示および予約するには、`--region` CLI 引数を使用してリージョンを us-west-2 に指定する必要があります。
- AWS SDK を使用して Hawaii 1 のコンタクトを一覧表示および予約するには、クライアントのリージョンを us-west-2 に設定する必要があります。この設定は、使用するプログラミング言語によって異なります。JavaScript を使用してこれを設定する方法の例については、AWS SDK for JavaScript のドキュメントを参照してください。詳細については、言語固有の SDK ドキュメントを参照してください。

AWS Ground Stationのセットアップ

使い始める前に AWS Ground Station、必要な AWS Identity and Access Management (IAM) 権限と、提供すべき宇宙船の認証情報を知っておく必要があります。次のステップに従ってアカウントをセットアップします。

トピック

サインアップして AWS アカウント

をお持ちでない場合は AWS アカウント、次の手順を実行して作成してください。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、管理ユーザーに管理アクセスを割り当て、ルートユーザーアクセスが必要なタスクを実行する場合にのみ、ルートユーザーを使用してください。

AWS サインアッププロセスが完了すると、確認メールが送信されます。https://aws.amazon.com/ の [マイアカウント] を選んで、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理できます。

管理ユーザーの作成

にサインアップしたら AWS アカウント、日常的なタスクに root ユーザーを使用しないように AWS IAM Identity Center、管理ユーザーを保護し、有効にしてから作成してください AWS アカウントのルートユーザー。

セキュリティを確保してください。 AWS アカウントのルートユーザー

1. [Root user] を選択し、 AWS アカウント メールアドレスを入力して、AWS Management Consoleアカウント所有者としてにログインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの「ルートユーザーとしてサインインする」を参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、IAM ユーザーガイドの「AWS アカウント root ユーザー (コンソール) の仮想 MFA デバイスを有効にする」を参照してください。

管理ユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。
2. IAM アイデンティティセンターで、管理ユーザーに管理アクセス権を付与します。

   IAM アイデンティティセンターディレクトリ をアイデンティティソースとして使用するチュートリアルについては、『ユーザーガイド』の「 IAM アイデンティティセンターディレクトリデフォルトでのユーザーアクセスの設定」を参照してください。AWS IAM Identity Center

管理ユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの「 AWS アクセスポータルへのサインイン」を参照してください。

AWS アカウントにGround Station 権限を追加

AWS Ground Station 管理ユーザーを必要とせずに使用するには、 AWS 新しいポリシーを作成してアカウントにアタッチする必要があります。

1. AWS Management Console にサインインし、IAM コンソールを開きます。
2. 新規ポリシーを作成します。以下のステップを使用します。

   a. ナビゲーションペインで、[Policies (ポリシー)] を選択し、次に [Create Policy (ポリシーの作成)] を選択します。

   b. [JSON] タブで、次のいずれかの値を使用して JSON を編集します。アプリケーションに最適な JSON を使用します。

      - Ground Station 管理者権限の場合は、次のように [アクション] を [groundstation:*] に設定します。

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

      - 読み取り専用権限の場合、次に示すように、[アクション] を oundstation:Get*、groundstation:List*、および groundstation:Describe* に設定します。

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

```
        "groundstation:List*",
        "groundstation:Describe*"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

- 多要素認証によるセキュリティを強化するには、以下のように [アクション] を groundstation: * に、[条件/ブール] を aws:: true に設定します。MultiFactorAuthPresent

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

3. IAM コンソールで、作成したポリシーを目的のユーザーにアタッチします。

IAM ユーザーの作成とポリシーのアタッチの詳細については、IAM ユーザーガイドを参照してください。

顧客オンボーディング

AWS Ground Station アカウントの登録を完了するには、 AWS Ground Station コンソールページの「サテライトとリソース」セクションでオンボーディングの詳細を確認してください。 AWS Ground Station チームがお客様と協力して、お客様のサテライトをサービスにオンボーディングします。衛星をオンボーディングすると、衛星を使用してコンタクトを管理できます。コンタクトを管理する手順については、「コンタクトの一覧表示と予約」で説明しています。

衛星をオンボーディングすると、衛星との間でデータを送受信するためのアクセス権が付与されます。ユーザーは、独自の衛星をオンボーディングするほかに、以下の衛星をオンボーディングし、AWS Ground Station を使用してダイレクトブロードキャストデータをダウンリンクすることもできます。

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

オンボーディングが完了すると、これらのサテライトにはすぐにアクセスして使用できるようになります。 AWS Ground Station サービスを簡単に使い始めることができるように、 AWS CloudFormation あらかじめ設定されたテンプレートが多数用意されています。このテンプレートにアクセスして使用する手順と詳細は、ユーザーガイドの「AWS CloudFormation テンプレートを使用してリソースを作成する」セクションに記載されています。

これらの衛星および送信されるデータタイプの詳細については、Aqua、JPSS-1/NOAA-20 および SNPP、および Terra を参照してください。

次のステップ

AWS Ground Station アカウントの設定が完了し、設定する準備が整いました。「開始方法」に進んで、 AWS Ground Stationを使用するようにリソースを設定します。

の開始方法 AWS Ground Station

AWS Ground Station を使用すると、衛星からデータをコマンド、制御、ダウンリンクできます。

を使用すると AWS Ground Station、グラウンドステーションアンテナへのアクセスを 1 分単位でスケジュールし、アンテナ時間に対してのみ料金を支払うことができます。 AWS Ground Station は、コンタクトデータをアカウントの Amazon Simple Storage Service (Amazon S3) バケットに非同期で配信するか、アカウントの Amazon Elastic Compute Cloud (Amazon EC2) インスタンスとの間でストリーミングすることで同期的に配信します。次のステップでは、Amazon S3 バケット内でコンタクトデータを非同期的に受信するために必要なリソースを設定する方法について説明します。Amazon EC2 へのデータ配信の使用方法の詳細については、Amazon EC2 へのデータ配信 ガイドを参照してください。

トピック

基本概念

開始する前に、「」の基本概念を理解しておく必要があります AWS Ground Station。詳細については、「コアコンポーネント」を参照してください。

次に、前提条件「」に進み、 の使用を開始するための前提条件について学習します AWS Ground Station。

前提条件

の使用を開始する前に AWS Ground Station、適切な認証情報を持つ AWS アカウントがあることを確認してください。「AWS Ground Stationのセットアップ」の手順を実行します。

> **Note**
>
> ブロードバンド DigIF データ配信を使用する場合、手順については「AWS Ground Station エージェントユーザーガイド」を参照してください。

それ以外の場合は、「ステップ 1: AWS CloudFormation テンプレートを選択する」に進みます。

ステップ 1: AWS CloudFormation テンプレートを選択する

衛星をオンボードしたら、ミッションプロファイルを定義して、衛星からデータをダウンリンクする AWS Ground Station アンテナ設定を定義する必要があります。このプロセスを支援するために、公開ブロードキャスト衛星を使用するナローバンドとワイドバンド DigIF データ配信の両方用に事前設定された AWS CloudFormation テンプレートを提供しています。これらのテンプレートを使用すると、 の使用を簡単に開始できます AWS Ground Station。の詳細については AWS CloudFormation、「AWS とは」を参照してください CloudFormation。

受け取るコンタクトのタイプに応じて、以下のリストから適切な CFN テンプレートタイプを選択します。

- ナローバンド S3 データ配信 AWS CloudFormation テンプレート.
- 広帯域 DigIf S3 データ配信 AWS CloudFormation テンプレート.

事前に作成された AWS CloudFormation テンプレートのいずれかを使用しない場合は、「」の手順を参照してください独自のテンプレートの構築。

ナローバンド S3 データ配信 AWS CloudFormation テンプレート

事前設定済みテンプレート

現在、コンタクトごとに S3 バケットへの複数のデータストリームを設定できます。これらのデータストリームは、2 つの異なる形式で使用できます。VITA-49 Signal/IP データを含むデータストリームは、最大 54 MHz の帯域幅の S バンド信号および X バンド信号に対して設定できます。VITA-49 拡張データ/IP は、最大 500 MHz の帯域幅の復調および/またはデコードされた X バンド信号に対して設定できます。

AWS Ground Station には、サービスの使用方法を示す両方のデータストリーム形式のテンプレートが用意されています。このガイドを使用して、適切なテンプレートを見つけてください。

利用可能なテンプレート

事前設定されたテンプレートを使用して、Aqua、SNPP、JPSS-1/NOAA-20、および Terra 衛星からダイレクトブロードキャストデータを受信できます。これらのAWS CloudFormationテンプレートには、コンタクトをスケジュール AWS Ground Station して実行し、アカウントの

Amazon S3 バケットでデータを受信するために必要な と Amazon S3 リソースが含まれています。Aqua、SNPP、JPSS-1/NOAA-20、および Terra がアカウントにオンボーディングされていない場合は、「カスタマーオンボーディング」を参照してください。

ナローバンドデータ配信テンプレート

問い合わせにナローバンドデータ配信を使用している場合は、以下の AWS CloudFormation テンプレートを使用します。

- という名前の AWS CloudFormation テンプレートには、コンタクトをスケジュールし、復調およびデコードされたダイレクトブロードキャストデータを受信するために必要な Amazon S3 バケットと AWS Ground Station リソースAquaSnppJpss-1DemodDecodeS3DataDelivery.ymlが含まれています。このテンプレートは、NASA Direct Readout Labs ソフトウェア (RT-STPS および IPOPP) を使用してデータを処理する予定の場合に適しています。

  を使用してテンプレートをダウンロードするには AWS CLI、次のコマンドを使用します。

  ```
  aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
  AquaSnppJpss-1DemodDecodeS3DataDelivery.yml .
  ```

  ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

  ```
  https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-
  us-west-2/AquaSnppJpss-1DemodDecodeS3DataDelivery.yml
  ```

  次のリンク AWS CloudFormation を使用して、 でテンプレートを直接指定できます。

  ```
  https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
  AquaSnppJpss-1DemodDecodeS3DataDelivery.yml
  ```

- という名前の AWS CloudFormation テンプレートには、コンタクトをスケジュールし、VITA-49 Signal/IP ダイレクトブロードキャストデータを受信するために必要な Amazon S3 バケットと AWS Ground Station リソースAquaSnppJpss-1TerraDigIfS3DataDelivery.ymlが含まれています。 VITA-49 このテンプレートは、後処理の前にデータを復調および複合するためにソフトウェア定義無線 (SDR) を使用してデータを処理する場合に適切な開始ポイントとして使用できます。

  を使用してテンプレートをダウンロードするには AWS CLI、次のコマンドを使用します。

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
AquaSnppJpss-1TerraDigIfS3DataDelivery.yml .
```

ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-
us-west-2/AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

次のリンク AWS CloudFormation を使用して、 でテンプレートを直接指定できます。

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

テンプレートによって定義されるリソース

両方のテンプレートには同じリソースが含まれています。唯一の違いはアンテナ設定です。詳細については、次の「アンテナ設定」を参照してください。

- Amazon S3 バケット - ダウンリンクされたデータの配信先となるバケット。このバケットの名前は、「S3 記録設定」に示されている条件を満たすために `aws-groundstation` で始まります。
- IAM ロール - ダウンリンクされたデータを Amazon S3 バケットに書き込むときに が AWS Ground Station 引き受ける`groundstation.amazonaws.com`サービスプリンシパルによって引き受け可能なロール。
- Amazon S3 バケットポリシー - IAM ロールが Amazon S3 バケットとそのオブジェクトで以下のアクションを実行することを許可するポリシー。
  - `s3:GetBucketLocation`
  - `s3:PutObject`
- 追跡設定 - アンテナシステムが衛星を上空を通過するときに衛星を追跡する方法を定義する AWS Ground Station 追跡設定。
- S3 録画設定 - データを配信するときに が使用する Amazon AWS Ground Station S3 バケットと IAM ロールを参照する S3 録画設定。 Amazon S3 AWS Ground Station
- アンテナ設定 - コンタクト中に AWS Ground Station アンテナを設定する方法を指定する AWS Ground Station アンテナ設定。`AquaSnppJpss-1DemodDecodeS3DataDelivery.yml` テンプレートには、ダウンリンクされたデータを Amazon S3 バケットに配信する前に復調およびデ

コードするようにアンテナを設定するアンテナダウンリンク復調デコード設定が含まれています。 AWS Ground Station Amazon S3 `AquaSnppJpss-1TerraDigIfS3DataDelivery.yml` 代わりに、 には、VITA-49 Signal/IP パケットとしてデータを Amazon S3 に配信するように AWS Ground Station アンテナを設定するアンテナダウンリンク設定が含まれています。

- ミッションプロファイル - すべての AWS Ground Station 設定をグループ化し、参照されている設定を使用してコンタクトをスケジュールおよび実行できるようにする AWS Ground Station ミッションプロファイル。

広帯域 DigIf S3 データ配信 AWS CloudFormation テンプレート

ブロードバンド DigIF データ配信テンプレート

問い合わせに Wideband Digital intermediate Frequency (DigIF) データ配信を使用している場合は、以下の AWS CloudFormation テンプレートを使用します。

- という名前の AWS CloudFormation テンプレート`DirectBroadcastSatelliteWbDigIfS3DataDelivery.yml`には、コンタクトをスケジュールし、エージェント経由で VITA-49 Signal/IP ダイレクトブロードキャストデータを受信するために必要な Amazon S3 バケットと AWS Ground Station リソースが含まれています。 VITA-49 AWS Ground Station このテンプレートは、後処理の前にデータを復調および複合するためにソフトウェア定義無線 (SDR) を使用してデータを処理する場合に適切な開始ポイントとして使用できます。 AWS Ground Station エージェントの詳細については、「」を参照してくださいAWS Ground Station エージェントユーザーガイド。

  を使用してテンプレートをダウンロードするには AWS CLI、次のコマンドを使用します。

  ```
  aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/agent/s3_recording/
  DirectBroadcastSatelliteWbDigIfS3DataDelivery.yml .
  ```

  ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

  ```
  https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-
  us-west-2/agent/s3_recording/DirectBroadcastSatelliteWbDigIfS3DataDelivery.yml
  ```

  次のリンク AWS CloudFormation を使用して、 でテンプレートを直接指定できます。

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
agent/s3_recording/DirectBroadcastSatelliteWbDigIfS3DataDelivery.yml
```

このテンプレートで定義するリソース

- Amazon S3 バケット - ダウンリンクされたデータの配信先となるバケット。このバケットの名前は、「S3 記録設定」に示されている条件を満たすために aws-groundstation で始まります。
- IAM ロール - ダウンリンクされたデータを Amazon S3 バケットに書き込むときに が AWS Ground Station 引き受けるgroundstation.amazonaws.comサービスプリンシパルによって引き受け可能なロール。
- Amazon S3 バケットポリシー - IAM ロールが Amazon S3 バケットとそのオブジェクトで以下のアクションを実行することを許可するポリシー。
  - s3:GetBucketLocation
  - s3:PutObject
- AWS KMS キー - データフローの暗号化に使用される AWS KMS キー。
- Ground Station キーロール - AWS KMS キーにアクセスして使用してデータフローを復号するために AWS Ground Station が引き受ける IAM ロール
- Ground Station キーアクセスポリシー - データ配信キーに対して AWS Ground Station 実行できるアクションを定義する IAM ポリシー
- 追跡設定 - アンテナシステムが衛星を上空を通過するときに衛星を追跡する方法を定義する AWS Ground Station 追跡設定。
- S3 録画設定 - データを配信するときに が使用する Amazon AWS Ground Station S3 バケットと IAM ロールを参照する S3 録画設定。 Amazon S3 AWS Ground Station
- Aqua、SNPP、JPSS-1/NOAA-20、Terra の Antenna Configs - Aqua、SNPP、JPSS-1/NOAA-20、Terra とのコンタクト中に AWS Ground Station アンテナを設定する方法を指定する 3 つの個別のアンテナ設定。 AWS Ground Station テンプレートには、VITA-49 Signal/IP パケットとしてデータを Amazon S3 に配信するように AWS Ground Station アンテナを設定するアンテナダウンリンク設定が含まれています。
- Aqua、SNPP、JPSS-1/NOAA-20、Terra のミッションプロファイル - Aqua、SNPP、JPSS-1/NOAA-20、Terra で参照される設定を使用してコンタクトをスケジュールおよび実行できるように、すべての AWS Ground Station 設定をグループ化する 3 つの個別の AWS Ground Station ミッションプロファイル。

独自のテンプレートの構築

独自の衛星のコンタクトをスケジュールして実行するようにリソースを設定するには、衛星の設定と一致するようにアカウント内の AWS Ground Station リソースを設定する必要があります。この作業をお客様自身で行うのは困難です。 AWS Ground Station チームは、衛星からダウンリンクおよびアップリンクするようにアカウント内の AWS Ground Station リソースを設定するのに役立ちます。で使用する独自の衛星を設定するには AWS Ground Station、AWS Support にお問い合わせください。

ステップ 2: AWS CloudFormation スタックを設定する

ユースケースに最適なテンプレートを選択したら、 AWS CloudFormation スタックを設定します。この手順で作成されるリソースは、作成時のリージョンに設定されます。

1. で AWS Management Console、サービス > CloudFormation を選択します。
2. ナビゲーションペインで、[Stacks] を選択します。次に、[スタックの作成] - [With new resources (standard)] の順に選択します。
3. [スタックの作成] ページで、次のいずれかを実行して、「the section called “ステップ 1: AWS CloudFormation テンプレートを選択する”」で選択したテンプレートを指定します。

   a. テンプレートソースとして [Amazon S3 URL] を選択し、Amazon S3 URL で使用するテンプレートの URL をコピーして貼り付けます。[次へ] を選択します。

   b. テンプレートソースとして [テンプレートファイルをアップロード] を選択し、[ファイルを選択] を選択します。「the section called “ステップ 1: AWS CloudFormation テンプレートを選択する”」でダウンロードしたテンプレートをアップロードします。[次へ] を選択します。
4. [Specify stack details] (スタックの詳細を指定) ページで以下の手順を実行します。

   a. [スタックの名前] ボックスに名前を入力します。将来のエラーの可能性を減らすために、単純な名前を使用することをお勧めします。

   b. [次へ] を選択します。
5. Amazon EC2 インスタンスのスタックオプションと詳細オプションを設定します。

   a. [タグ] セクションと [アクセス権限] セクションでタグとアクセス権限を追加します。

   b. [スタックポリシー]、[ロールバック設定]、[通知オプション]、および [スタック作成オプション] を変更します。

c. [次へ] を選択します。

6. スタックの詳細を確認したら、[CAPABILITY] 確認を選択し、[スタックの作成] を選択します。

AWS Ground Station エージェントユーザーガイド

トピック

概要

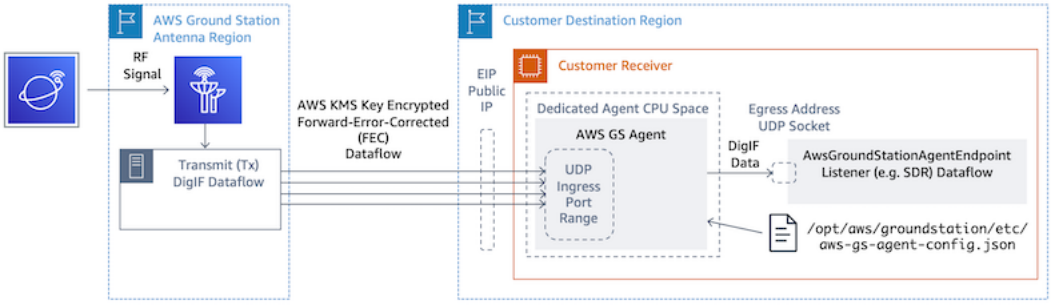
AWS Ground Station エージェントとは

AWS Ground Station Agent は RPM として提供されており、 AWS Ground Station お客様は AWS Ground Station との通信中に同期広帯域デジタル中間周波数 (DigIF) データフローを受信 (ダウンリンク) できます。お客様はデータ配信用に次の 2 つのオプションを選択できます。

1. EC2 インスタンスへのデータ配信-顧客所有の EC2 インスタンスへのデータ配信。 AWS Ground Station AWS Ground Station エージェントは顧客が管理します。このオプションは、ほぼリアルタイムのデータ処理が必要な場合に最適です。EC2 データ配信の詳細については、「Amazon EC2 へのデータ配信 ガイド」を参照してください。
2. S3 バケットへのデータ配信 - Ground Station マネージドサービスを介して、お客様が所有する AWS S3 バケットへのデータ配信。S3 データ配信の詳細については、「の開始方法 AWS Ground Stationガイド」を参照してください。

どちらのデータ配信モードでも、お客様は一連の AWS リソースを作成する必要があります。信頼性、正確性、サポート性を確保するために、 CloudFormation テンプレートを使用して AWS リソースを作成することを強くお勧めします。各コンタクトは EC2 または S3 のいずれかにのみデータを配信でき、両方に同時に配信することはできません。

> **Note**
>
> S3 データ配信は Ground Station マネージドサービスであるため、このガイドでは EC2 インスタンスへのデータ配信に焦点を当てています。



ソフトウェア定義無線 (SDR) または同様のリスナーを使用して、 AWS Ground Station アンテナリージョンから EC2 インスタンスへの DigIF データフロー。

エージェントの機能 AWS Ground Station

AWS Ground Station エージェントはデジタル中間周波数 (DigIF) ダウンリンクデータを受信し、復号化されたデータを出力します。これにより、以下が可能になります。

- 40 MHz から 400 MHz の帯域幅までの DigIF ダウンリンク機能。
- AWS ネットワーク上の任意のパブリック IP (AWS Elastic IP) への高レート、低ジッターの DigIF データ配信。
- 前方誤り訂正 (FEC) による信頼性の高いデータ配信。
- 顧客が管理する暗号化キーを使用してデータを安全に配信します。 AWS KMS

エージェントの要件

> **Note**
>
> AWS Ground Station このエージェントガイドは、ガイドを使用してGGround Station にオンボーディングしたことを前提としています。AWS Ground Stationのセットアップ

AWS Ground Station エージェントレシーバーの EC2 インスタンスには、DigIF データをエンドポイントに確実かつ安全に配信するための一連の AWS リソースが必要です。

1. EC2 レシーバーを起動する VPC。
2. データの暗号化/復号化用の AWS KMS キー。
3. SSM セッションマネージャー用に設定された SSH キーまたは EC2 インスタンスプロファイル。
4. 以下のことを許可するネットワーク/セキュリティグループのルール:
    1. AWS Ground Station データフローエンドポイントグループで指定されたポートからの UDP トラフィック。エージェントは、入力データフローエンドポイントにデータを配信するために使用される一連の連続したポートを予約します。
    2. インスタンスへの SSH アクセス (注: または AWS セッションマネージャーを使用して EC2 インスタンスにアクセスすることもできます)。
    3. エージェント管理用の、パブリックにアクセス可能な S3 バケットへの読み取りアクセス。
    4. エージェントがサービスと通信できるようにするポート 443 の SSL トラフィック。 AWS Ground Station
    5. AWS Ground Station `com.amazonaws.global.groundstation`管理対象プレフィックスリストからのトラフィック。

さらに、パブリックサブネットを含む VPC 設定が必要です。サブネット設定の背景情報については、「VPC ユーザーガイド」を参照してください。

互換性のある設定:

1. パブリックサブネットの EC2 インスタンスに関連付けられた Elastic IP。
2. (任意のサブネットで) EC2 インスタンスにアタッチされ、パブリックサブネットの ENI インスタンスに関連付けられた Elastic IP。

EC2 インスタンスと同じセキュリティグループを使用することも、少なくとも以下の最小ルールセットを含むセキュリティグループを指定することもできます。

- AWS Ground Station データフローエンドポイントグループで指定されたポートからの UDP トラフィック。

これらのリソースがあらかじめ設定されている AWS CloudFormation EC2 データ配信テンプレートの例の「ワイドバンド DigIF データ配信テンプレート」セクションを参照してください。テンプレートの選択

VPC の図

図: パブリックサブネットの EC2 インスタンスに関連付けられた Elastic IP

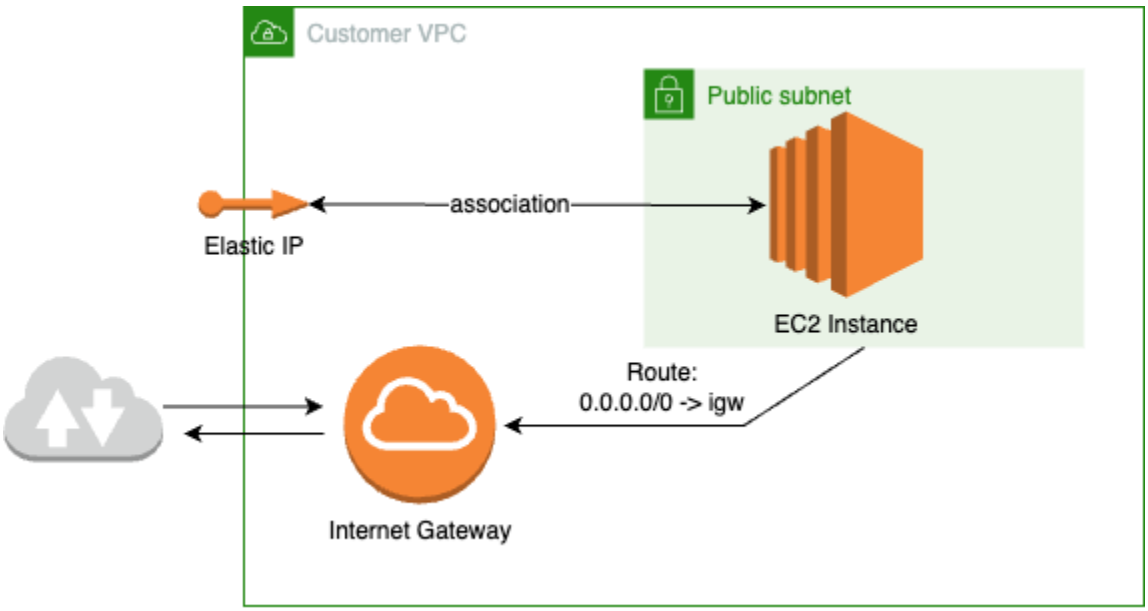
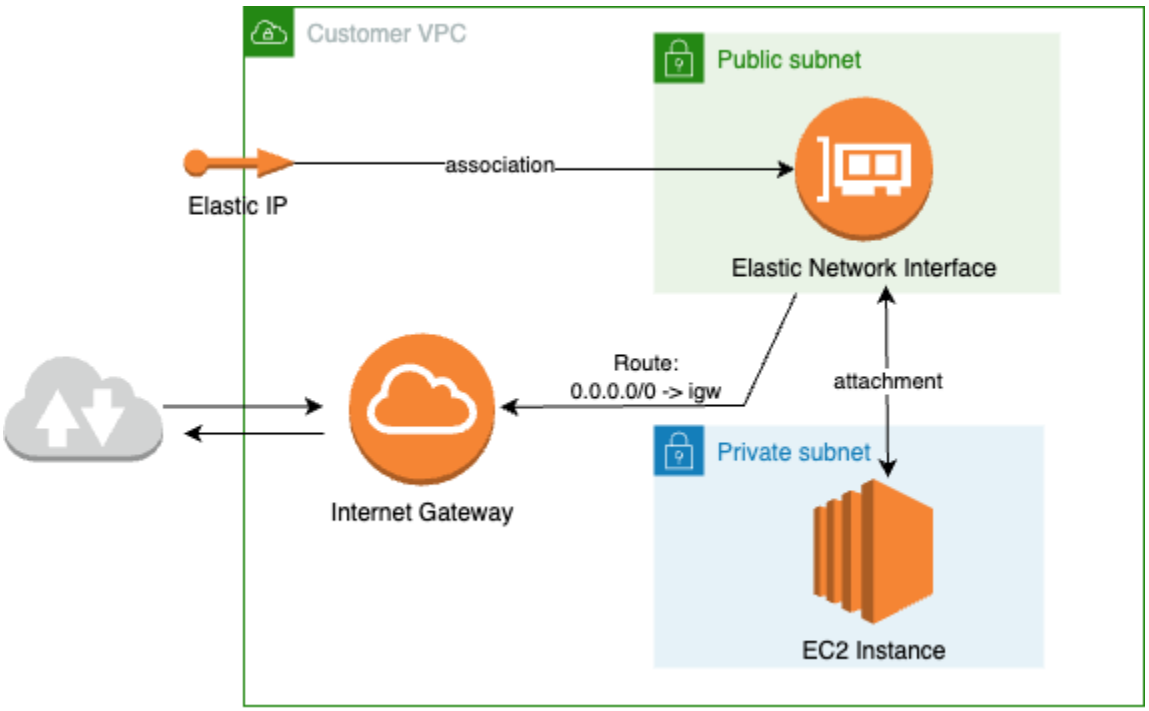


図: プライベートサブネットで EC2 インスタンスにアタッチされ、パブリックサブネットの ENI に関連付けられた Elastic IP



サポートされるオペレーティングシステム

Amazon Linux 2 (5.10+ カーネル)

サポートされているインスタンスタイプは以下のとおりです。EC2 インスタンスの選択と CPU プランニング

AWS Ground Station エージェントによるデータ配信

以下の図は、広帯域デジタル中間周波数 (DigIF) AWS Ground Station コンタクト中にデータがどのように流れるかの概要を示しています。

AWS Ground Station エージェントは連絡先のデータプレーンコンポーネントのオーケストレーションを行います。コンタクトをスケジュールする前に、エージェントを正しく設定、起動し、に登録 (エージェントの起動時に自動的に登録されます) する必要があります。 AWS Ground Stationさらに、データ受信ソフトウェア (ソフトウェア定義の無線など) が稼働していて、AwsGroundStationAgentEndpointEgressAddress でデータを受信するように設定されている必要があります。

AWS Ground Station AWS Ground Station エージェントはバックグラウンドでタスクを受信し、AWS KMS 転送中に適用された暗号化を取り消してから、ソフトウェア定義無線 (SDR) が受信して

いる宛先エンドポイント EgressAddress にデータを転送します。 AWS Ground Station エージェントとその基盤となるコンポーネントは、インスタンスで実行されている他のアプリケーションのパフォーマンスに影響を与えないように、設定ファイルに設定された CPU 境界を尊重します。

顧客は、コンタクトに関係するレシーバーインスタンスで AWS Ground Station Agent を実行させる必要があります。お客様がすべてのデータフローを単一のレシーバーインスタンスで受信することを希望する場合、以下に示すように、1 つの AWS Ground Station Agent で複数のデータフローをオーケストレーションできます。

複数のデータフロー、単一のレシーバー

シナリオの例:

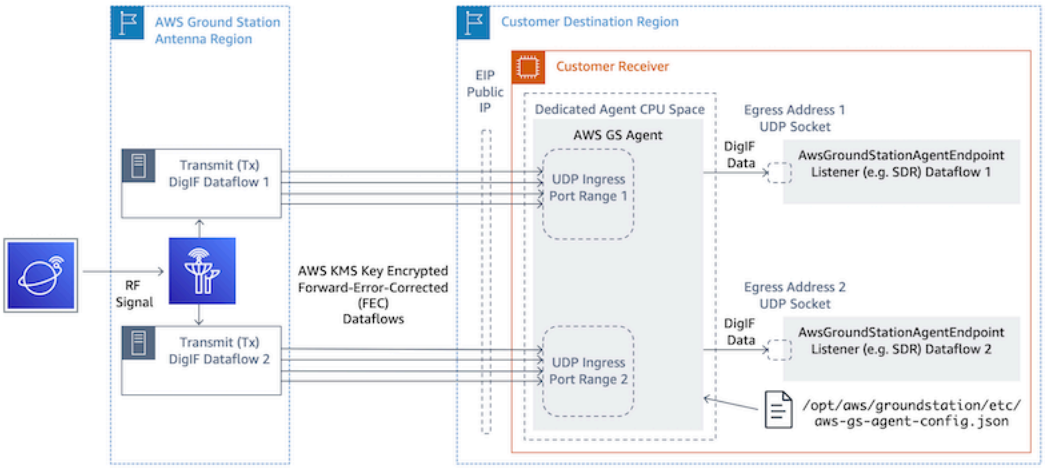
お客様が、同じ EC2 レシーバーインスタンスで DigIF データフローとして 2 つのアンテナダウンリンクを受信するとします。2 つのダウンリンクは 200 MHz と 100 MHz とします。

AwsGroundStationAgentEndpoints:

各データフローに 1 つずつ、合計 2 つの `AwsGroundStationAgentEndpoint` リソースがあります。両方のエンドポイントには同じパブリック IP アドレス (`ingressAddress.socketAddress.name`) が割り当てられます。データフローは同じ EC2 インスタンスで受信されるため、イングレスの `portRange` は重複しないようにしてください。どちらの `egressAddress.socketAddress.port` も一意である必要があります。

CPU プランニング:

- AWS Ground Station インスタンス上で単一のエージェントを実行するための1コア (2vCPU)。
- DigIF データフロー 1 (テーブル内の 200MHz ルックアップ) を受信するための 6 コア (12 vCPU)。CPU コアプランニング
- 4 コア (8 vCPU) で DigIF データフロー 2 (テーブル内の 100 MHz ルックアップ) を受信します。CPU コアプランニング
- 専用エージェントの合計 CPU 容量 = 同じソケット上の 11 コア (22 vCPU)。



複数のデータフロー、複数のレシーバー

シナリオの例:

お客様が、異なる EC2 レシーバーインスタンスで DigIF データフローとして、2 つのアンテナダウンリンクを受信するとします。どちらのダウンリンクも 400 MHz とします。

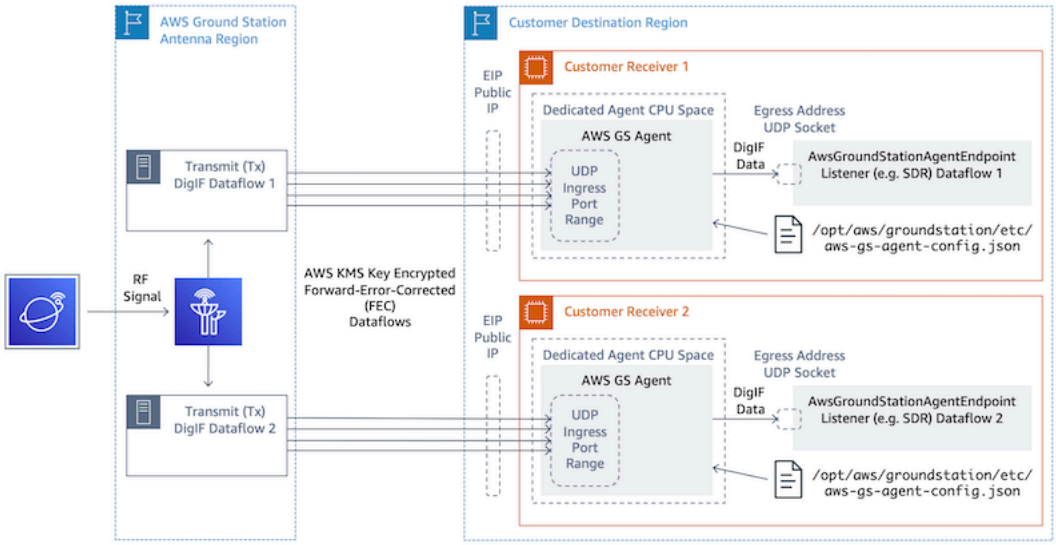
AwsGroundStationAgentEndpoints:

各データフローに 1 つずつ、合計 2 つの `AwsGroundStationAgentEndpoint` リソースがあります。エンドポイントには異なるパブリック IP アドレス (`ingressAddress.socketAddress.name`) が割り当てられます。データフローは別のインフラストラクチャで受信され、互いに競合しないため、`ingressAddress` または `egressAddress` のいずれのポート値にも制限はありません。

CPU プランニング:

- レシーバーインスタンス 1
  - AWS Ground Station インスタンス上で単一のエージェントを実行するための1コア (2vCPU)。
  - 9 コア (18 vCPU) で DigIF データフロー 1 (テーブル内の 400MHz ルックアップ) を受信します。CPU コアプランニング
  - 専用エージェントの合計 CPU 容量 = 同じソケット上の 10 コア (20 vCPU)。
- レシーバーインスタンス 2
  - AWS Ground Station インスタンス上で単一のエージェントを実行するための1コア (2vCPU)。
  - 9 コア (18 vCPU) で DigIF データフロー 2 (テーブル内の 400MHz ルックアップ) を受信します。CPU コアプランニング

- 専用エージェントの合計 CPU 容量 = 同じソケット上の 10 コア (20 vCPU)。



EC2 インスタンスの選択と CPU プランニング

サポートされる EC2 インスタンスタイプ

処理負荷の高いデータ配信ワークフローのため、 AWS Ground Station Agent の動作には専用の CPU コアが必要です。次のインスタンスタイプがサポートされています。どのインスタンスタイプがお客様のユースケースに最も適しているかを判断するには、「CPU コアプランニング」を参照してください。

インスタンスタイプ	デフォルト vCPU	デフォルトの CPU コア
c5.12xlarge	48	24
c5.18xlarge	72	36
c5.24xlarge	96	48
c5n.18xlarge	72	36
c5n.metal	72	36
c6i.32xlarge	128	64

インスタンスタイプ	デフォルト vCPU	デフォルトの CPU コア
g4dn.12xlarge	48	24
g4dn.16xlarge	64	32
g4dn.metal	96	48
m4.16xlarge	64	32
m5.12xlarge	48	24
m5.24xlarge	96	48
m6i.32xlarge	128	64
p3dn.24xlarge	96	48
p4d.24xlarge	96	48
r5.24xlarge	96	48
r5.metal	96	48
r5n.24xlarge	96	48
r5n.metal	96	48
r6i.32xlarge	128	64

CPU コアプランニング

AWS Ground Station Agent には、データフローごとに L3 キャッシュを共有する専用のプロセッサコアが必要です。エージェントは、ハイパースレッド (HT) CPU ペアを利用するように設計されているため、使用するには HT ペアを予約する必要があります。ハイパースレッドペアは、1 つのコアに含まれる仮想 CPU (vCPU) のペアです。次の表は、データフローのデータレートと、1 つのデータフローでエージェントに予約されている必要なコア数とのマッピングを示しています。この表は Cascade Lake 以降の CPU を想定しており、サポートされているどのインスタンスタイプでも有効です。帯域幅が表のエントリ間の場合は、次に高いものを選択してください。

エージェントは管理と調整のために追加のリザーブドコアを必要とするため、必要なコアの合計は、各データフローに必要なコア (下の図を参照) に 1 つの追加コア (vCPUs 2) を加えたものになります。

AntennaDownlink 帯域幅 (MHz)	予想される VITA-49.2 DigIF データレート (メガバイト/秒)	コア数 (HT CPU ペア)	仮vCPU 合計
50	1,000	3	6
100	2000	4	8
150	3000	5	10
200	4000	6	12
250	5000	6	12
300	6000	7	14
350	7000	8	16
400	8000	9	18

アーキテクチャ情報の収集

`lscpu`システムのアーキテクチャに関する情報を提供します。基本出力には、どの vCPUs (「CPU」というラベルが付いている) がどの NUMA ノードに属しているか (各 NUMA ノードが L3 キャッシュを共有している) かがわかります。以下では、`c5.24xlarge`エージェントの設定に必要な情報を収集するためにインスタンスを調べます。 AWS Ground Station これには、vCPUs の数、コア、vCPU とノード間の関連付けなどの有用な情報が含まれます。

```
> lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 96
On-line CPU(s) list: 0-95
Thread(s) per core: 2             <------
Core(s) per socket: 24
```

```
Socket(s): 2
NUMA node(s): 2
Vendor ID: GenuineIntel
CPU family: 6
Model: 85
Model name: Intel(R) Xeon(R) Platinum 8275CL CPU @ 3.00GHz
Stepping: 7
CPU MHz: 3601.704
BogoMIPS: 6000.01
Hypervisor vendor: KVM
Virtualization type: full
L1d cache: 32K
L1i cache: 32K
L2 cache: 1024K
L3 cache: 36608K
NUMA node0 CPU(s): 0-23,48-71     <------
NUMA node1 CPU(s): 24-47,72-95    <------
```

AWS Ground Station エージェント専用のコアには、割り当てられた各コアの両方の vCPUs が必要です。データフローのすべてのコアは同じ NUMA ノードに存在する必要があります。`-plscpu`コマンドのオプションを使用すると、エージェントの設定に必要なコアと CPU を関連付けることができます。関連するフィールドは、CPU (vCPU)、Core、L3 (そのコアがどの L3 キャッシュを共有しているかを示す) です。ほとんどの Intel プロセッサでは NUMA ノードは L3 キャッシュと同じであることに注意してください。

a `lscpu -p` の出力の次のサブセットを考えてみましょう `c5.24xlarge` (わかりやすくするために省略してフォーマットしています)。

```
CPU,Core,Socket,Node,,L1d,L1i,L2,L3
0   0    0      0     0   0   0  0
1   1    0      0     1   1   1  0
2   2    0      0     2   2   2  0
3   3    0      0     3   3   3  0
...
16  0    0      0     0   0   0  0
17  1    0      0     1   1   1  0
18  2    0      0     2   2   2  0
19  3    0      0     3   3   3  0
```

出力から、コア 0 にはvCPUs 0 と 16 が含まれ、コア 1 にはvCPUs 1 と 17 が、コア 2 にはvCPUs 2 と 18 が含まれていることがわかります。つまり、ハイパースレッドのペアは 0 と 16、1 と 17、2 と 18 です。

CPU 割り当ての例

例として、350 MHz `c5.24xlarge` の二重極性広帯域ダウンリンク用のインスタンスを使用します。の表から、350 MHzのダウンリンクでは、単一のデータフローに8コア (16個のvCPUs) が必要であることがわかります。CPU コアプランニングつまり、2 つのデータフローを使用するこの二重極性セットアップでは、エージェント用に合計 16 コア (32 vCPUs) と 1 コア (2 vCPUs) が必要です。

とを含むの出力はわかっています。`lscpu` `c5.24xlarge` `NUMA node0 CPU(s): 0-23,48-71` `NUMA node1 CPU(s): 24-47,72-95`NUMA node0 には必要以上のものがあるため、0 ~ 23 と 48-71 のコアからのみ割り当てます。

まず、L3 キャッシュまたは NUMA ノードを共有するデータフローごとに 8 つのコアを選択します。次に、`lscpu -p`出力内の対応する vCPUs (「CPU」というラベル) を検索します。付録:c5.24xlarge の出力 (全文`lscpu -p`)コア選択プロセスの例は以下のようになります。

- コア 0 ~ 1 は OS 用に予約してください。
- フロー 1: vCPUs 2 ~ 9 と 50-57 にマッピングするコア 2 ~ 9 を選択します。
- フロー 2: vCPUs 10-17 と 58-65 にマップするコア 10 ~ 17 を選択します。
- エージェントコア:vCPUs 18 と 66 にマップするコア 18 を選択します。

この結果、vCPUs が 2 ~ 18 と 51-66 になるため、エージェントを提供するリストはになります。`[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66]`で説明されているように、これらの CPU 上で独自のプロセスが実行されていないことを確認する必要があります。AWS Ground Station エージェントと並行してサービスとプロセスを実行する

この例で選択されている特定のコアはいくぶん任意であることに注意してください。データフローごとに L3 キャッシュをすべて共有するという要件を満たしている限り、他のコアセットでも機能します。

付録:c5.24xlarge の出力 (全文`lscpu -p`)

```
> lscpu -p
```

```
# The following is the parsable format, which can be fed to other
# programs. Each different item in every column has an unique ID
# starting from zero.
# CPU,Core,Socket,Node,,L1d,L1i,L2,L3
0,0,0,0,,0,0,0,0
1,1,0,0,,1,1,1,0
2,2,0,0,,2,2,2,0
3,3,0,0,,3,3,3,0
4,4,0,0,,4,4,4,0
5,5,0,0,,5,5,5,0
6,6,0,0,,6,6,6,0
7,7,0,0,,7,7,7,0
8,8,0,0,,8,8,8,0
9,9,0,0,,9,9,9,0
10,10,0,0,,10,10,10,0
11,11,0,0,,11,11,11,0
12,12,0,0,,12,12,12,0
13,13,0,0,,13,13,13,0
14,14,0,0,,14,14,14,0
15,15,0,0,,15,15,15,0
16,16,0,0,,16,16,16,0
17,17,0,0,,17,17,17,0
18,18,0,0,,18,18,18,0
19,19,0,0,,19,19,19,0
20,20,0,0,,20,20,20,0
21,21,0,0,,21,21,21,0
22,22,0,0,,22,22,22,0
23,23,0,0,,23,23,23,0
24,24,1,1,,24,24,24,1
25,25,1,1,,25,25,25,1
26,26,1,1,,26,26,26,1
27,27,1,1,,27,27,27,1
28,28,1,1,,28,28,28,1
29,29,1,1,,29,29,29,1
30,30,1,1,,30,30,30,1
31,31,1,1,,31,31,31,1
32,32,1,1,,32,32,32,1
33,33,1,1,,33,33,33,1
34,34,1,1,,34,34,34,1
35,35,1,1,,35,35,35,1
36,36,1,1,,36,36,36,1
37,37,1,1,,37,37,37,1
38,38,1,1,,38,38,38,1
39,39,1,1,,39,39,39,1
```

```
40,40,1,1,,40,40,40,1
41,41,1,1,,41,41,41,1
42,42,1,1,,42,42,42,1
43,43,1,1,,43,43,43,1
44,44,1,1,,44,44,44,1
45,45,1,1,,45,45,45,1
46,46,1,1,,46,46,46,1
47,47,1,1,,47,47,47,1
48,0,0,0,,0,0,0,0
49,1,0,0,,1,1,1,0
50,2,0,0,,2,2,2,0
51,3,0,0,,3,3,3,0
52,4,0,0,,4,4,4,0
53,5,0,0,,5,5,5,0
54,6,0,0,,6,6,6,0
55,7,0,0,,7,7,7,0
56,8,0,0,,8,8,8,0
57,9,0,0,,9,9,9,0
58,10,0,0,,10,10,10,0
59,11,0,0,,11,11,11,0
60,12,0,0,,12,12,12,0
61,13,0,0,,13,13,13,0
62,14,0,0,,14,14,14,0
63,15,0,0,,15,15,15,0
64,16,0,0,,16,16,16,0
65,17,0,0,,17,17,17,0
66,18,0,0,,18,18,18,0
67,19,0,0,,19,19,19,0
68,20,0,0,,20,20,20,0
69,21,0,0,,21,21,21,0
70,22,0,0,,22,22,22,0
71,23,0,0,,23,23,23,0
72,24,1,1,,24,24,24,1
73,25,1,1,,25,25,25,1
74,26,1,1,,26,26,26,1
75,27,1,1,,27,27,27,1
76,28,1,1,,28,28,28,1
77,29,1,1,,29,29,29,1
78,30,1,1,,30,30,30,1
79,31,1,1,,31,31,31,1
80,32,1,1,,32,32,32,1
81,33,1,1,,33,33,33,1
82,34,1,1,,34,34,34,1
83,35,1,1,,35,35,35,1
```

```
84,36,1,1,,36,36,36,1
85,37,1,1,,37,37,37,1
86,38,1,1,,38,38,38,1
87,39,1,1,,39,39,39,1
88,40,1,1,,40,40,40,1
89,41,1,1,,41,41,41,1
90,42,1,1,,42,42,42,1
91,43,1,1,,43,43,43,1
92,44,1,1,,44,44,44,1
93,45,1,1,,45,45,45,1
94,46,1,1,,46,46,46,1
95,47,1,1,,47,47,47,1
```

エージェントのインストール

AWS Ground Station Agent は以下の方法でインストールできます。

1. AWS CloudFormation テンプレート (推奨)。
2. Amazon EC2 に手動でインストールします。

CloudFormation テンプレートを使用する

EC2 CloudFormation データ配信テンプレートは、EC2 インスタンスにデータを配信するために必要な AWS リソースを作成します。 AWS CloudFormation このテンプレートは、 AWS Ground Station Agent AWS Ground Station がプリインストールされているマネージド AMI を使用します。次に、作成された EC2 インスタンスの起動スクリプトがエージェント設定ファイルにデータを入力し、必要なパフォーマンスチューニング (EC2 インスタンスのパフォーマンスチューニング) を適用します。

ステップ 1: AWS リソースを作成する

テンプレート ダイレクトブロードキャスト衛星ブロードバンド DigIF テンプレート (ワイドバンド) を使用して AWS リソースを作成します。

ステップ 2: エージェントステータスを確認する

デフォルトでは、エージェントは設定され、アクティブ (開始) になっています。エージェントのステータスを確認するには、EC2 インスタンス (SSH または SSM セッションマネージャー) に接続して、AWS Ground Station エージェントステータス を表示できます。

EC2 に手動でインストールする

GGround Station CloudFormation ではテンプレートを使用してAWS リソースをプロビジョニングすることを推奨していますが、標準テンプレートでは不十分な場合もあります。このような場合は、ニーズに合わせてテンプレートをカスタマイズすることをお勧めします。それでも要件を満たさない場合は、AWS リソースを手動で作成し、エージェントをインストールできます。

ステップ 1: AWS リソースを作成する

コンタクトに必要な AWS リソースを手動で設定する手順については、「手動でリソースを作成および構成する」を参照してください。

AwsGroundStationAgentEndpointこのリソースは、 AWS Ground Station エージェント経由で DigIF データフローを受信するためのエンドポイントを定義するもので、連絡を正常に取るうえで不可欠です。API ドキュメントは API リファレンスに記載されていますが、このセクションでは Agent に関連する概念について簡単に説明します。 AWS Ground Station

`ingressAddress`エンドポイントは、 AWS Ground Station AWS KMS エージェントがアンテナから暗号化された UDP トラフィックを受信する場所です。`socketAddress name` は、(アタッチされた EIP からの) EC2 インスタンスのパブリック IP です。`portRange` は、他の使用時間から予約されている範囲内に 300 個以上の連続したポートである必要があります。手順については、「入力ポートの予約 - ネットワークに影響あり」を参照してください。これらのポートは、レシーバーインスタンスが実行されている VPC のセキュリティグループで UDP イングレストラフィックを許可するように設定する必要があります。

エンドポイントの `egressAddress` は、エージェントが DigIF データフローをお客様に引き渡す場所です。お客様は、この場所の UDP ソケットを経由してデータを受信するアプリケーション (SDR など) を用意する必要があります。

ステップ 2: EC2 インスタンスを作成する

以下の AMI がサポートされています。

1. AWS Ground Station AMI (* は AMI が作成された日付) `groundstation-al2-gs-agent-ami-*` にはエージェントがインストールされています (推奨)。
2. `amzn2-ami-kernel-5.10-hvm-x86_64-gp2`.

ステップ 2: エージェントをダウンロードしてインストールする

> **Note**
>
> AWS Ground Station 前のステップでエージェント AMI を選択しなかった場合は、このセクションのステップを完了する必要があります。

エージェントをダウンロードする

AWS Ground Station エージェントはリージョン固有の S3 バケットから入手でき、AWS コマンドライン (CLI) を使用してサポートされている EC2 インスタンスにダウンロードできます。$ {AWS::Region} は、サポートされている AWS Ground Station `s3://groundstation-wb-digif-software-${AWS::Region}/aws-groundstation-agent/latest/amazon_linux_2_x86_64/aws-groundstation-agent.rpm` コンソールおよびデータ配信リージョンのいずれかを指します。

例: 最新の rpm バージョンを AWS リージョン us-east-2 から /tmp フォルダにローカルにダウンロードするとします。

```
aws s3 --region us-east-2 cp s3://groundstation-wb-digif-software-us-east-2/aws-groundstation-agent/latest/amazon_linux_2_x86_64/aws-groundstation-agent.rpm /tmp
```

特定のバージョンの AWS Ground Station Agent をダウンロードする必要がある場合は、S3 バケットのバージョン固有のフォルダからダウンロードできます。

例: rpm のバージョン 1.0.2716.0 を AWS リージョン us-east-2 から /tmp フォルダにローカルにダウンロードするとします。

```
aws s3 --region us-east-2 cp s3://groundstation-wb-digif-software-us-east-2/aws-groundstation-agent/1.0.2716.0/amazon_linux_2_x86_64/aws-groundstation-agent.rpm /tmp
```

> **Note**
>
> ダウンロードした RPM が販売元であることを確認するには AWS Ground Station、の指示に従ってください。RPM のインストールの検証

エージェントをインストールする

```
sudo yum install ${MY_RPM_FILE_PATH}

Example: Assumes agent is in the "/tmp" directory
sudo yum install /tmp/aws-groundstation-agent.rpm
```

ステップ 4: エージェントを設定する

エージェントをインストールしたら、エージェント設定ファイルを更新する必要があります。エージェントの設定 を参照してください。

ステップ 5: パフォーマンスチューニングを適用する

AWS Ground Station エージェント AMI: AWS Ground Station 前のステップでエージェント AMI を選択した場合は、次のパフォーマンスチューニングを適用します。

- ハードウェア割り込みと受信キューのチューニング - CPU とネットワークに影響あり
- 入力ポートの予約 - ネットワークに影響あり
- 再起動

その他の AMI: 前のステップで他の AMI を選択した場合は、「EC2 インスタンスのパフォーマンスチューニング」に記載されているすべてのチューニングを適用し、インスタンスを再起動します。

ステップ 6: エージェントを管理する

エージェントの起動、停止、およびステータスの確認については、「エージェントを管理する」を参照してください。

エージェントを管理する

AWS Ground Station Agent には、組み込みの Linux コマンドツールを使用して、エージェントを設定、起動、停止、アップグレード、ダウングレード、アンインストールするための以下の機能があります。

トピック

AWS Ground Station エージェント設定

に移動すると/opt/aws/groundstation/etc、 aws-gs-agent-config .json という名前の 1 つのファイルが含まれているはずです。「エージェント設定ファイル」を参照

AWS Ground Station エージェントスタート

```
#start
sudo systemctl start aws-groundstation-agent

#check status
systemctl status aws-groundstation-agent
```

エージェントがアクティブであることを示す出力を生成する必要があります。

```
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
```

```
Active: active (running) since Tue 2023-03-14 00:39:08 UTC; 1 day 13h ago
Docs: https://aws.amazon.com/ground-station/
Main PID: 8811 (aws-gs-agent)
CGroup: /system.slice/aws-groundstation-agent.service
##8811 /opt/aws/groundstation/bin/aws-gs-agent production
```

AWS Ground Station エージェントストップ

```
#stop
sudo systemctl stop aws-groundstation-agent

#check status
systemctl status aws-groundstation-agent
```

エージェントが非アクティブ (停止中) であることを示す出力を生成する必要があります。

```
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: inactive (dead) since Thu 2023-03-09 15:35:08 UTC; 6min ago
Docs: https://aws.amazon.com/ground-station/
Process: 84182 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=0/SUCCESS)
Main PID: 84182 (code=exited, status=0/SUCCESS)
```

AWS Ground Station エージェントアップグレード

1. エージェントの最新バージョンをダウンロードします。エージェントをダウンロードする を参照してください。
2. エージェントを停止します。

```
#stop
sudo systemctl stop aws-groundstation-agent

#confirm inactive (stopped) state
```

```
systemctl status aws-groundstation-agent
```

3. エージェントを更新します。

```
sudo yum update ${MY_RPM_FILE_PATH}

# check the new version has been installed correctly by comparing the agent version
 with the starting agent version
yum info aws-groundstation-agent

# reload the systemd configuration
sudo systemctl daemon-reload

# restart the agent
sudo systemctl restart aws-groundstation-agent

# check agent status
systemctl status aws-groundstation-agent
```

AWS Ground Station エージェントダウングレード

1. 必要なエージェントバージョンをダウンロードします。エージェントをダウンロードする を参照してください。
2. エージェントをダウングレードします。

```
# get the starting agent version
yum info aws-groundstation-agent

# stop the agent service
sudo systemctl stop aws-groundstation-agent

# downgrade the rpm
sudo yum downgrade ${MY_RPM_FILE_PATH}

# check the new version has been installed correctly by comparing the agent version
 with the starting agent version
yum info aws-groundstation-agent
```

```
# reload the systemd configuration
sudo systemctl daemon-reload

# restart the agent
sudo systemctl restart aws-groundstation-agent

# check agent status
systemctl status aws-groundstation-agent
```

AWS Ground Station エージェントアンインストール

エージェントをアンインストールすると、/opt/aws/groundstation/etc/ .json の名前が aws-gs-agent-config /opt/aws/groundstation/etc/ .json.rpmsave に変更されます。aws-gs-agent-configエージェントを同じインスタンスに再度インストールすると、 aws-gs-agent-config .json のデフォルト値が書き込まれるため、AWS リソースに対応する正しい値で更新する必要があります。エージェント設定ファイル を参照してください。

```
sudo yum remove aws-groundstation-agent
```

AWS Ground Station エージェントステータス

エージェントのステータスは、アクティブ (エージェントが実行中) か、非アクティブ (エージェントが停止中) のいずれかです。

```
systemctl status aws-groundstation-agent
```

出力例には、エージェントがインストール済み、非アクティブ (停止中)、有効 (起動時にサービスを開始) のステータスが表示されます。

```
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
```

```
Active: inactive (dead) since Thu 2023-03-09 15:35:08 UTC; 6min ago
Docs: https://aws.amazon.com/ground-station/
Process: 84182 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=0/SUCCESS)
Main PID: 84182 (code=exited, status=0/SUCCESS)
```

AWS Ground Station エージェント RPM 情報

```
yum info aws-groundstation-agent
```

出力は次のとおりです。

> **Note**
>
> 「バージョン」は、エージェントが公開している最新のバージョンによって異なる場合があります。

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : aws-groundstation-agent
Arch        : x86_64
Version     : 1.0.2677.0
Release     : 1
Size        : 51 M
Repo        : installed
Summary     : Client software for AWS Ground Station
URL         : https://aws.amazon.com/ground-station/
License     : Proprietary
Description : This package provides client applications for use with AWS Ground Station
```

エージェントの設定

エージェントをインストールしたら、/opt/aws/groundstation/etc/aws-gs-agent-config.json でエージェント設定ファイルを更新する必要があります。

エージェント設定ファイル

例

```
{
  "capabilities": [
    "arn:aws:groundstation:eu-central-1:123456789012:dataflow-endpoint-group/
bb6c19ea-1517-47d3-99fa-3760f078f100"
  ],
  "device": {
  "privateIps": [
    "127.0.0.1"
  ],
  "publicIps": [
    "1.2.3.4"
  ],
  "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
}
```

フィールド内訳

機能

機能は、データフローエンドポイントグループの Amazon リソースネームとして指定されます。

必須: True

形式: 文字列配列

- 値: 機能 ARN → 文字列

例:

```
"capabilities": [
    "arn:aws:groundstation:${AWS::Region}:${AWS::AccountId}:dataflow-endpoint-group/
${DataflowEndpointGroupId}"
]
```

デバイス

このフィールドには、現在の EC2「デバイス」を列挙するのに必要な追加フィールドが含まれています。

必須: True

フォーマット: オブジェクト

メンバー:

- privateIps
- publicIps
- agentCpuCores
- networkAdapters

privateIps

このフィールドは現在使用されていませんが、今後のユースケースに備えて含まれています。値が含まれていない場合、デフォルトで [“127.0.0.1”] になります。

必須: False

形式: 文字列配列

- 値: IP アドレス → 文字列

例：

```
"privateIps": [
    "127.0.0.1"
],
```

publicIps

データフローエンドポイントグループごとの Elastic IP (EIP)。

必須: True

形式: 文字列配列

- 値: IP アドレス → 文字列

例：

```
"publicIps": [
    "9.8.7.6"
],
```

agentCPUCores

aws-gs-agent どの仮想コアをプロセス用に予約するかを指定します。この値を適切に設定するための要件については、「CPU コアプランニング」を参照してください。

必須: True

形式: 整数配列

- 値: コア数 → 整数

例：

```
"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
]
```

networkAdapters

これは、データを受信するイーサネットアダプタ (つまり ENI) に接続されたインターフェイスに対応します。

必須: False

形式: 文字列配列

- 値: イーサネットアダプタの名前 (`ifconfig` を実行すると検索できます)

例：

```
"networkAdapters": [
    "eth0"
]
```

EC2 インスタンスのパフォーマンスチューニング

> **Note**
>
> CloudFormation テンプレートを使用して AWS リソースをプロビジョニングした場合、これらの調整は自動的に適用されます。AMI を使用した場合、または EC2 インスタンスを手動で作成した場合、最も信頼性の高いパフォーマンスを実現するには、これらのパフォーマンスチューニングを適用する必要があります。

チューニングを適用した後は、必ずインスタンスを再起動してください。

トピック

- ハードウェア割り込みと受信キューのチューニング - CPU とネットワークに影響あり
- Rx 割り込み合体のチューニング - ネットワークに影響あり
- Rx リングバッファのチューニング - ネットワークに影響あり
- CPU C ステートのチューニング - CPU に影響あり
- 入力ポートの予約 - ネットワークに影響あり
- 再起動

ハードウェア割り込みと受信キューのチューニング - CPU とネットワークに影響あり

このセクションでは、systemd、SMP IRQ、受信パケットステアリング (RPS)、受信フローステアリング (RFS) の CPU コア使用率を設定します。使用しているインスタンスタイプに基づく一連の推奨設定については、「付録:割り込み/RPS チューニングの推奨パラメータ」を参照してください。

1. systemd プロセスをエージェント CPU コアから切り離します。

2. ハードウェア割り込みリクエストをエージェント CPU コアから離してルーティングします。
3. 1 つのネットワークインターフェイスカードのハードウェアキューがネットワークトラフィックのボトルネックにならないように RPS を設定します。
4. CPU キャッシュヒットレートを高め、ネットワーク遅延を低減するように RFS を設定します。

RPM が提供する `set_irq_affinity.sh` スクリプトは、上記のすべてを自動的に構成します。crontab に追加して、起動のたびに適用されるようにします。

```
echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh
 '${interrupt_core_list}' '${rps_core_mask}' >> /var/log/user-data.log 2>&1" >>/var/
spool/cron/root
```

- カーネルと OS `interrupt_core_list` 専用のコアに置き換えます。通常は 1 番目と 2 番目のコアと、ハイパースレッド対応のコアペアです。これと、上記で選択したコアとが重複しないようにしてください。(例:ハイパースレッドの 96 CPU インスタンスの場合は '0,1,48,49')。
- `rps_core_mask` は、受信パケットを処理する CPU を指定する 16 進数のビットマスクで、各桁は 4 個の CPU を表します。また、右から 8 文字ごとにカンマで区切る必要があります。すべての CPU を許可し、キャッシュにバランシングを任せることをお勧めします。
  - 各インスタンスタイプに推奨されるパラメータのリストについては、「付録:割り込み/RPS チューニングの推奨パラメータ」を参照してください。
- 96-CPU インスタンスの例:

```
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
```

Rx 割り込み合体のチューニング - ネットワークに影響あり

割り込み合体により、ホストシステムに大量の割り込みが発生するのを防ぎ、ネットワークのスループットを向上させることができます。この構成では、パケットが収集され、128 マイクロ秒ごとに 1 つの割り込みが発生します。crontab に追加して、起動のたびに適用されるようにします。

```
echo "@reboot sudo ethtool -C ${interface} rx-usecs 128 tx-usecs 128 >>/var/log/user-
data.log 2>&1" >>/var/spool/cron/root
```

- interface を、データを受信するように設定されたネットワークインターフェイス (イーサネットアダプタ) に置き換えます。通常、これは eth0 です。なぜなら、これは、EC2 インスタンスに割り当てられるデフォルトのネットワークインターフェイスであるためです。

Rx リングバッファのチューニング - ネットワークに影響あり

Rx リングバッファのリングエントリ数を増やして、バースト接続中のパケットドロップやオーバーランを防止します。crontab に追加して、起動するたびに正しく設定されるようにします。

```
echo "@reboot sudo ethtool -G ${interface} rx 16384 >>/var/log/user-data.log 2>&1" >>/
var/spool/cron/root
```

- interface を、データを受信するように設定されたネットワークインターフェイス (イーサネットアダプタ) に置き換えます。通常、これは eth0 です。なぜなら、これは、EC2 インスタンスに割り当てられるデフォルトのネットワークインターフェイスであるためです。
- c6i.32xlarge インスタンスを設定する場合、リングバッファを、16384 の代わりに 8192 に設定するようにコマンドを変更する必要があります。

CPU C ステートのチューニング - CPU に影響あり

CPU C ステートを設定して、コンタクトの開始時にパケットが失われる原因となるアイドリングを防止します。インスタンスの再起動が必要です。

```
echo "GRUB_CMDLINE_LINUX_DEFAULT=\"console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 intel_idle.max_cstate=1
 processor.max_cstate=1 max_cstate=1\"" >/etc/default/grub
echo "GRUB_TIMEOUT=0" >>/etc/default/grub
grub2-mkconfig -o /boot/grub2/grub.cfg
```

入力ポートの予約 - ネットワークに影響あり

カーネルの使用状況と競合しないように、AwsGroundStationAgentEndpoint の入力アドレスのポート範囲内のすべてのポートを予約します。ポートの使用が競合すると、コンタクトやデータ配信の失敗につながります。

```
echo "net.ipv4.ip_local_reserved_ports=${port_range_min}-${port_range_max}" >> /etc/
sysctl.conf
```

- 例えば、echo "net.ipv4.ip_local_reserved_ports=42000-43500" >> /etc/sysctl.conf などです。

再起動

すべてのチューニングが正常に適用されたら、インスタンスを再起動してチューニングを有効にします。

```
sudo reboot
```

付録:割り込み/RPS チューニングの推奨パラメータ

このセクションでは、チューニングセクション「ハードウェア割り込みと受信キューのチューニング - CPU とネットワークに影響あり」セクションで使用するための推奨パラメータ値を決定します。

ファミリー	インスタンスタイプ	${interrupt_core_list}	${rps_core_mask}
C6i	• c6i.32xlarge	• 0,1,64,65	• ffffffff, ffffffff, ffffffff, ffffffff

ファミリー	インスタンスタイプ	${interrupt_core_list}	${rps_core_mask}
c5	• c5.24xlarge • c5.18xlarge • c5.12xlarge	• 0,1,48,49 • 0,1,36,37 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ff、ffffff ff、ffffffff • ffff,ffffffff
c5n	• c5n.metal • c5n.18xlarge	• 0,1,36,37 • 0,1,36,37	• ff、ffffff ff、ffffffff • ff、ffffff ff、ffffffff
m5	• m5.24xlarge • m5.12xlarge	• 0,1,48,49 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ffff,ffffffff
r5	• r5.metal • r5.24xlarge	• 0,1,48,49 • 0,1,48,49	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff, ffffffff
r5n	• r5n.metal • r5n.24xlarge	• 0,1,48,49 • 0,1,48,49	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff, ffffffff

ファミリー	インスタンスタイプ	${interrupt_core_list}	${rps_core_mask}
g4dn	• g4dn.metal • g4dn.16xlarge • g4dn.12xlarge	• 0,1,48,49 • 0,1,32,33 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff • ffff,ffffffff
p4d	• p4d.24xlarge	• 0,1,48,49	• ffffffff, ffffffff, ffffffff
p3dn	• p3dn.24xlarge	• 0,1,48,49	• ffffffff, ffffffff, ffffffff

DigIF へのコンタクトの実行を準備をする

1. 必要なデータフローに関して CPU コアプランニングを確認し、エージェントが使用できるコアのリストを提供します。CPU コアプランニング を参照してください。
2. エージェント設定ファイルを確認します。 AWS Ground Station AWS Ground Station エージェント設定 を参照してください。
3. 必要なパフォーマンスチューニングが適用されていることを確認します。EC2 インスタンスのパフォーマンスチューニング を参照してください。
4. 記載されているベスト・プラクティスをすべて守っていることを確認してください。ベストプラクティス を参照してください。
5. 次の方法で、予定されている連絡開始時間より前に AWS Ground Station Agent が起動していることを確認します。

```
systemctl status aws-groundstation-agent
```

6. 次の方法で、 AWS Ground Station 予定されている連絡開始時刻より前にエージェントが正常であることを確認します。

```
aws groundstation get-dataflow-endpoint-group --dataflow-endpoint-group-id
 ${DATAFLOW-ENDPOINT-GROUP-ID} --region ${REGION}
```

`awsGroundStationAgentEndpoint` の `agentStatus` がアクティブで、`auditResults` が正常であることを検証します。

ベストプラクティス

EC2 のベストプラクティス

EC2 の最新のベストプラクティスに従い、十分なデータストレージの可用性を確保します。

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-best-practices.html

Linux スケジューラ

Linux スケジューラは、対応するプロセスが特定のコアに固定されていない場合、UDP ソケット上のパケットを並べ替えることができます。UDP データを送受信するスレッドは、データ転送中は特定のコアに固定する必要があります。

AWS Ground Station 管理対象プレフィックスリスト

アンテナからの通信を許可するネットワークルールを指定するときは、`com.amazonaws.global.groundstation` という AWS 管理のプレフィックスリストを利用することをお勧めします。AWS マネージドプレフィックスリストの詳細については、「AWS マネージドプレフィックスリストの使用」を参照してください。

単一のコンタクトの制限

AWS Ground Station Agent は、コンタクトごとに複数のストリームをサポートしますが、一度に単一のコンタクトのみをサポートします。スケジュールの問題を防ぐため、複数のデータフローエンドポイントグループ間でインスタンスを共有しないでください。単一のエージェント設定が複数の異なる DFEG ARN に関連付けられている場合、登録に失敗します。

AWS Ground Station エージェントと並行してサービスとプロセスを実行する

エージェントと同じ EC2 インスタンスでサービスとプロセスを起動する場合、 AWS Ground Station AWS Ground Station エージェントと Linux カーネルで使用されていない vCPUs にそれらをバインドすることが重要です。これにより、ボトルネックが発生したり、通信中にデータが失われたりする可能性があります。特定の vCPUs にバインドするというこの概念は、アフィニティと呼ばれます。

避けるべきコア:

- agentCpuCores from エージェント設定ファイル
- interrupt_core_list (ハードウェア割り込みと受信キューのチューニング - CPU とネットワークに影響あり から)。
  - デフォルト値は、以下から確認できます。付録:割り込み/RPS チューニングの推奨パラメータ

例として、**c5.24xlarge**インスタンスを使います。

指定した場合

```
"agentCpuCores": [24,25,26,27,72,73,74,75]"
```

そして走った

```
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
```

次に、次のコアを避けてください。

```
0,1,24,25,26,27,48,49,72,73,74,75
```

アフィニッシュサービス (systemd)

新しく開始されたサービスは、前述のサービスに自動的にアフィニッシュされます。interrupt_core_list開始したサービスのユースケースでコアを追加する必要がある場合や、混雑の少ないコアが必要な場合は、このセクションに従ってください。

サービスが現在どのアフィニティに設定されているかをコマンドで確認します。

```
systemctl show --property CPUAffinity <service name>
```

空の値が表示される場合はCPUAffinity=、上記のコマンドのデフォルトコアを使用する可能性が高いということです。...bin/set_irq_affinity.sh <using the cores here> ...

特定のアフィニティをオーバーライドして設定するには、以下のコマンドを実行してサービスファイルの場所を検索します。

```
systemctl show -p FragmentPath <service name>
```

ファイルを開いて (vi、などを使用して) 変更しnano、CPUAffinity=<core list>[Service]次のようなセクションに記述します。

```
[Unit]
...

[Service]
...
CPUAffinity=2,3

[Install]
...
```

ファイルを保存してサービスを再起動し、以下とのアフィニティを適用します。

```
systemctl daemon-reload
systemctl restart <service name>

# Additionally confirm by re-running
systemctl show --property CPUAffinity <service name>
```

詳細については、Red Hat Enterprise Linux 8-カーネルの管理、監視、更新-第 27 章をご覧ください。systemd を使用して CPU アフィニティと NUMA ポリシーを設定する。

アフィニッシュプロセス (スクリプト)

Linux のデフォルト動作では、マシン上のどのコアでも使用できるようになるため、新しく起動したスクリプトやプロセスは手動でアフィニッシュすることを強くお勧めします。

実行中のプロセス (python や Bash スクリプトなど) のコアの競合を避けるには、以下のコマンドでプロセスを起動してください。

```
taskset -c <core list> <command>
# Example: taskset -c 8 ./bashScript.sh
```

プロセスがすでに実行中の場合は、pidoftop、psなどのコマンドを使用して特定のプロセスのプロセス ID (PID) を調べてください。PID を使うと、以下との現在の親和性を確認できます。

```
taskset -p <pid>
```

また、以下のように変更することもできます。

```
taskset -p <core mask> <pid>
# Example: taskset -p c 32392 (which sets it to cores 0xc -> 0b1100 -> cores 2,3)
```

タスクセットについての詳細は、taskset-Linux のマニュアルページを参照してください。

トラブルシューティング

エージェントの起動の失敗

AWS Ground Station Agent が起動しない理由はいくつかありますが、最も一般的なシナリオは、エージェント設定ファイルの設定ミスが原因と考えられます。エージェントを起動すると (「AWS Ground Station エージェントスタート」を参照)、次のようなステータスが表示される場合があります。

```
#agent is automatically retrying a restart
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: activating (auto-restart) (Result: exit-code) since Fri 2023-03-10 01:48:14
 UTC; 23s ago
Docs: https://aws.amazon.com/ground-station/
Process: 43038 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=101)
Main PID: 43038 (code=exited, status=101)

#agent has failed to start
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: failed (Result: start-limit) since Fri 2023-03-10 01:50:15 UTC; 13s ago
Docs: https://aws.amazon.com/ground-station/
Process: 43095 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=101)
Main PID: 43095 (code=exited, status=101)
```

トラブルシューティング

```
sudo journalctl -u aws-groundstation-agent | grep -i -B 3 -A 3 'Loading Config' | tail
 -6
```

これによる出力は、次のようになります。

```
launch-aws-gs-agent[43095]: Running with options Production(ProductionOptions
 { endpoint: None, region: None })
launch-aws-gs-agent[43095]: Loading Config
launch-aws-gs-agent[43095]: System has 96 logical cores
systemd[1]: aws-groundstation-agent.service: main process exited, code=exited,
 status=101/n/a
systemd[1]: Unit aws-groundstation-agent.service entered failed state.
```

「Loading Config」の後にエージェントを起動できない場合、これは、エージェント設定に問題があることを示しています。エージェント設定を検証するには、「エージェント設定ファイル」を参照してください。

AWS Ground Station エージェントログ

AWS Ground Station エージェントは、コンタクトの実行、エラー、およびヘルスステータスに関する情報を、エージェントを実行しているインスタンスのログファイルに書き込みます。インスタンスに手動で接続することで、ログファイルを表示できます。

エージェントログは以下の場所で確認できます。

```
/var/log/aws/groundstation
```

利用できるコンタクトがない

連絡先をスケジュールするには、 AWS Ground Station 正常なエージェントが必要です。次の方法で AWS Ground Station API get-dataflow-endpoint-group にクエリを実行して、 AWS Ground Station エージェントが起動して正常であることを確認してください。

```
aws groundstation get-dataflow-endpoint-group --dataflow-endpoint-group-id ${DATAFLOW-
ENDPOINT-GROUP-ID} --region ${REGION}
```

`awsGroundStationAgentEndpoint` の `agentStatus` がアクティブで、`auditResults` が正常であることを検証します。

サポート情報

AWS サポートを通じて Ground Station チームにお問い合わせください。

1. 影響を受けているコンタクトがあれば、`contact_id` を伝えてください。この情報がないと、AWS Ground Station チームは特定の連絡先を調査できません。
2. 既に実施したすべてのトラブルシューティング手順に関する詳細を提供してください。
3. トラブルシューティングガイダンスに記載されているコマンドの実行中に表示されたエラーメッセージがあれば、提供してください。

エージェントのリリースノート

最新のエージェントバージョン

バージョン 1.0.3555.0

リリース日:2024 年 3 月 27 日

Support 終了日:2024 年 8 月 31 日

RPM チェックサム:

- SHA256: `108f3aceb00e5af549839cd766c56149397e448a6e1e1429c89a9eebb6bc0fc1`
- MD5: `65b72fa507fb0af32651adbb18d2e30f`

変更:

- タスク起動時に、選択した実行バージョンの Agent メトリックを追加します。
- 設定ファイルのサポートを追加して、他の実行バージョンがある場合に特定の実行可能なバージョンを避けるようにしました。
- ネットワークとルーティングの診断機能を追加。
- セキュリティ機能の追加。
- 一部のメトリックレポートエラーがログファイルではなく stdout/journal に書き込まれる問題を修正しました。
- ネットワークに到達できないソケットエラーを正常に処理します。
- 送信元と送信先のエージェント間のパケット損失と遅延を測定します。
- aws-gs-datapipe バージョン 2.0 をリリースして、新しいプロトコル機能をサポートし、連絡先を新しいプロトコルに透過的にアップグレードできるようにしました。

非推奨のエージェントバージョン

バージョン 1.0.2942.0

リリース日:2023 年 6 月 26 日

Support 終了日:2024年5月31日

RPM チェックサム:

- SHA256: 7d94b642577504308a58bab28f938507f2591d4e1b2c7ea170b77bea97b5a9b6
- MD5: 661ff2b8f11aba5d657a6586b56e0d8f

変更:

- Agent RPM がディスク上で更新され、変更を有効にするために Agent の再起動が必要な場合の エラーログを追加しました。
- Agent ユーザーガイドのチューニング手順に従い、正しく適用されていることを確認するための ネットワークチューニング検証を追加しました。
- ログのアーカイブに関する Agent ログに誤った警告が表示されるバグを修正しました。
- パケットロス検出が改善されました。
- Agent のインストールを更新して、Agent が既に実行中の場合に RPM をインストールまたはアップグレードできないようにしました。

バージョン 1.0.2716.0

リリース日:2023 年 3 月 15 日

Support 終了日:2024年5月31日

RPM チェックサム:

- SHA256: cb05b6a77dfcd5c66d81c0072ac550affbcefefc372cc5562ee52fb220844929
- MD5: 65266490c4013b433ec39ee50008116c

変更:

- タスク処理中に Agent に障害が発生した場合のログのアップロードを有効にします。
- 提供されているネットワークチューニングスクリプトの Linux 互換性バグを修正。

バージョン 1.0.2677.0

リリース日:2023 年 2 月 15 日

Support 終了日:2024年5月31日

RPM チェックサム:

- SHA256: 77cfe94acb00af7ca637264b17c9b21bd7afdc85b99dffdd627aec9e99397489
- MD5: b8533be7644bb4d12ab84de21341adac

変更:

- 初めて一般公開された Agent リリース。

RPM のインストールの検証

最新の RPM バージョン、RPM から検証された MD5 ハッシュ、および sha256sum を使用した SHA256 ハッシュを以下に示します。これらの値を組み合わせることで、Ground Station Agent に使用されている RPM バージョンを検証できます。

最新のエージェントバージョン

バージョン 1.0.3555.0

リリース日:2024 年 3 月 27 日

Support 終了日:2024 年 8 月 31 日

RPM チェックサム:

- SHA256: 108f3aceb00e5af549839cd766c56149397e448a6e1e1429c89a9eebb6bc0fc1
- MD5: 65b72fa507fb0af32651adbb18d2e30f

変更:

- タスク起動時に、選択した実行バージョンの Agent メトリックを追加します。
- 設定ファイルのサポートを追加して、他の実行バージョンがある場合に特定の実行可能なバージョンを避けるようにしました。
- ネットワークとルーティングの診断機能を追加。
- セキュリティ機能の追加。
- 一部のメトリックレポートエラーがログファイルではなく stdout/journal に書き込まれる問題を修正しました。
- ネットワークに到達できないソケットエラーを正常に処理します。

- 送信元と送信先のエージェント間のパケット損失と遅延を測定します。
- aws-gs-datapipe バージョン 2.0 をリリースして、新しいプロトコル機能をサポートし、連絡先を新しいプロトコルに透過的にアップグレードできるようにしました。

RPM を検証する

この RPM のインストールを検証するために必要なツールは以下のとおりです。

- sha256sum
- rpm

Amazon Linux 2 では、どちらのツールもデフォルトで提供されています。これらのツールは、使用している RPM が正しいバージョンであることを検証するのに役立ちます。まず、S3 バケットから最新の RPM をダウンロードします (RPM のダウンロードの手順については「エージェントをダウンロードする」を参照してください)。このファイルがダウンロードされたら、いくつか確認すべき点があります。

- RPM ファイルの sha256sum を計算します。使用しているコンピューティングインスタンスのコマンドラインから以下のアクションを実行します。

  ```
  sha256sum aws-groundstation-agent.rpm
  ```

  この値を取得し、上の表と比較します。これは、ダウンロードされた RPM ファイルが、AWS Ground Station がお客様に対してベンダリングした、使用向けの有効なファイルであることを示しています。ハッシュが一致しない場合は、RPM をインストールせず、コンピューティングインスタンスから削除します。
- ファイルの MD5 ハッシュもチェックして、RPM が侵害されていないことを確認します。このために、次のコマンドを実行することで、RPM コマンドラインツールを使用します。

  ```
  rpm -Kv ./aws-groundstation-agent.rpm
  ```

  ここに記載されている MD5 ハッシュが、上の表にあるバージョンの MD5 ハッシュと同じであることを検証します。これらのハッシュの両方が AWS ドキュメント内に記載されているこの表と

照合して検証されれば、お客様はダウンロードおよびインストールされた RPM が、RPM の安全で、侵害のないバージョンであることを確認できます。

コンタクトの一覧表示と予約

AWS Ground Station コンソールまたは AWS CLIを使用することにより、衛星データの入力、アンテナ位置の識別、選択した衛星のアンテナ使用時間のスケジュールを行うことができます。スケジュールされた時間の 8 日前まで、コンタクト予約を見直し、キャンセル、再スケジュールできます。また、リザーブドミニッツ料金モデルを使用している場合は、 AWS Ground Station リザーブドミニッツ料金プランの詳細を表示できます。

AWS Ground Station クロスリージョンのデータ配信をサポートします。選択したミッションプロファイルの一部であるデータフローエンドポイント設定によって、データの配信先のリージョンが決まります。クロスリージョンデータ配信の使用方法の詳細については、「クロスリージョンのデータ配信サービスの使用」を参照してください。

コンタクトをスケジュールするには、リソースを設定する必要があります。リソースを設定していない場合は、「開始方法」を参照してください。

トピック

- Ground Station コンソールの使用
- 連絡先の予約と管理は AWS CLI

Ground Station コンソールの使用

AWS Ground Station コンソールを使用して、連絡先の予約を予約、表示、キャンセルできます。コンソールを使用するには、 AWS Ground Station コンソールを開いて [AWS Ground Station連絡先を今すぐ予約] を選択します。

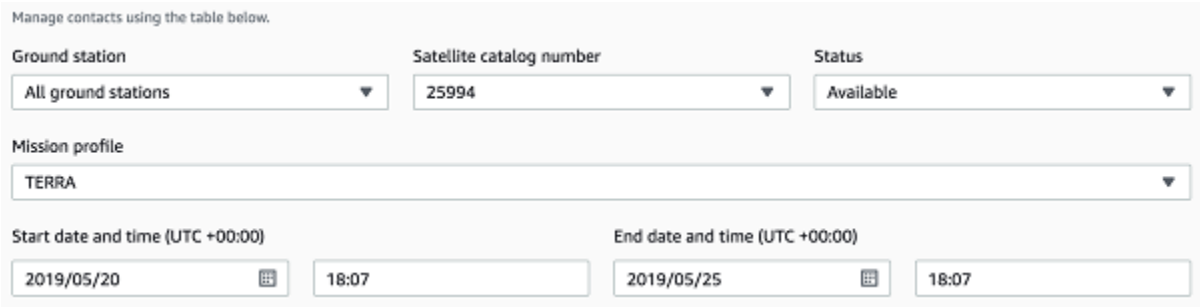
AWS Ground Station コンソールを使用して連絡先を予約、表示、キャンセルするには、以下のトピックを参照してください。

トピック

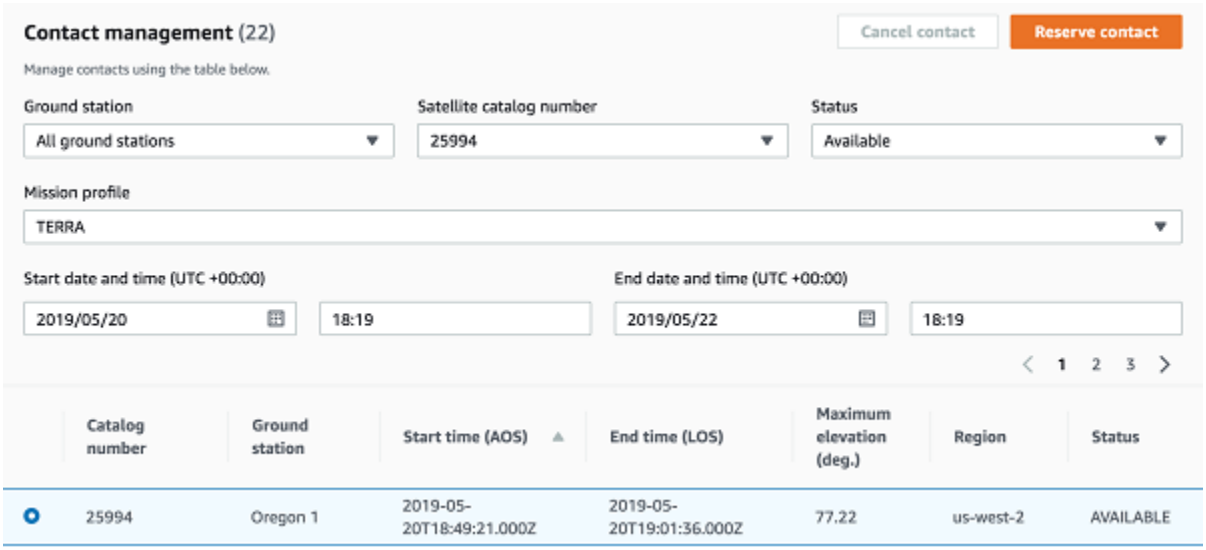
コンタクトを予約する

AWS Ground Station コンソールにアクセスしたら、設定したリソースを使って連絡先管理テーブルの連絡先を予約します。

1. [Contact management (コンタクトの管理)] 一覧で、使用可能なコンタクトの検索に使用するパラメータを選択します。[Status (ステータス)] フィルタを使用して [Available (使用可能)] のコンタクトが表示されるようにします。



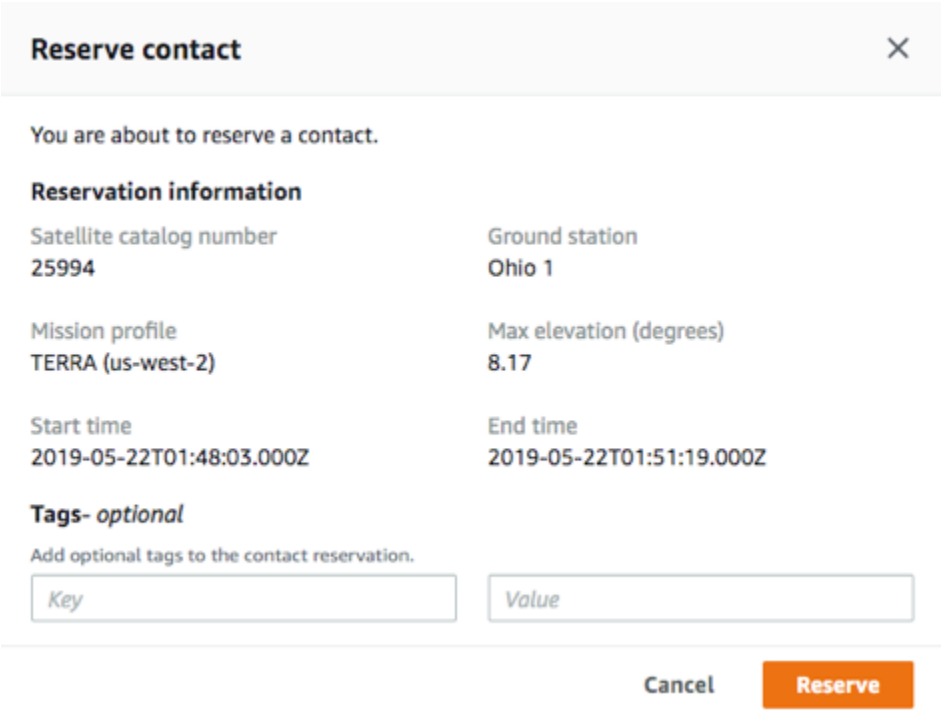
2. 要件を満たすコンタクトを選択して、[Reserve Contact (コンタクトを予約)] を選択します。



3. [コンタクトの予約] ダイアログボックスで、コンタクトの予約情報を確認します。

   a. (省略可能) [タグ] で、追加する各タグのキーと値を入力します。

   b. [予約] を選択します。

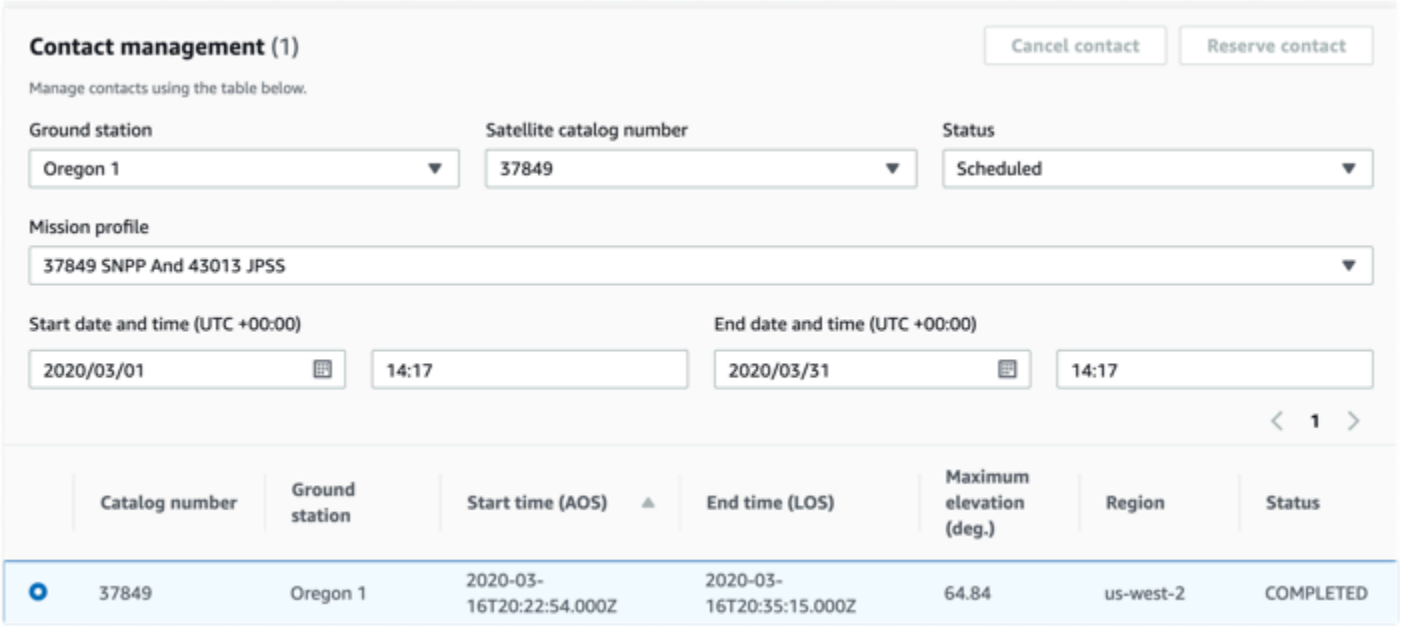


AWS Ground Station ミッションプロファイルの設定データを使用して、指定された地上局で連絡を取ります。

スケジュール済みのコンタクトと完了済みのコンタクトを表示する

連絡先をスケジュールすると、 AWS Ground Station コンソールを使用して予定された連絡と完了した連絡先の詳細を表示できます。

[Contact management (コンタクトの管理)] 一覧で、スケジュール済みのコンタクトと完了済みのコンタクトの検索に使用するパラメータを選択します。[Status (ステータス)] フィルターを使用して、[Scheduled (スケジュール済み)] または [Completed (完了)] のコンタクトを表示します。

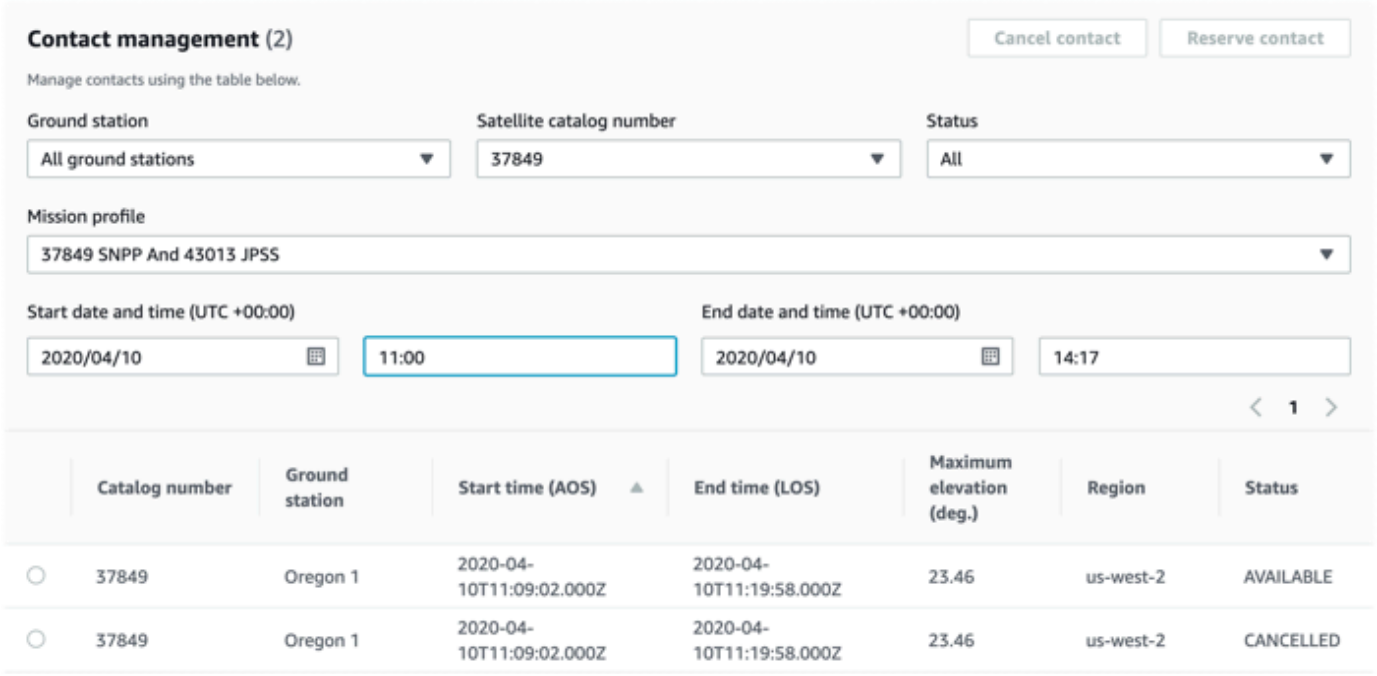


パラメータと一致したスケジュール済みまたは完了済みのコンタクトが一覧表示されます。

コンタクトのキャンセル

AWS Ground Station コンソールを使用して、スケジュールされた連絡先をキャンセルできます。

1. [Contact management (コンタクトの管理)] 一覧で、スケジュール済みのコンタクトと完了済みのコンタクトの検索に使用するパラメータを選択します。[Status (ステータス)] フィルターを使用して、[Scheduled (スケジュール済み)] のコンタクトを表示します。
2. スケジュール済みのコンタクトのリストから、キャンセルするコンタクトを選択します。次に、[Cancel Contact (コンタクトのキャンセル)] を選択します。
3. [Cancel contact (コンタクトのキャンセル)] ダイアログボックスで、[OK] を選択します。



コンタクトのステータスが [CANCELLED (キャンセル済み)] になります。

衛星の命名

AWS Ground Station コンソールには、連絡先ページを使用するときに、ユーザーが定義したサテライトの名前と Norad ID を表示する機能があります。衛星の名前を表示すると、スケジュール設定時に正しい衛星を選択しやすくなります。そのためには、タグを使用できます。

AWS Ground Station 衛星のタグ付けは、AWS CLI またはいずれかの AWS SDK を用いて tag-resource API 経由で行うことができます。このガイドでは、 AWS Ground Station CLI を使用して公共放送衛星Aqua (Norad ID 27424) にタグを付ける方法について説明します。`us-west-2`

AWS Ground Station CLI

AWS CLI を使用して通信できます。 AWS Ground Station AWS CLI を使用してサテライトをタグ付けする前に、 AWS CLI 以下の前提条件を満たす必要があります。

- インストールされていることを確認してください。 AWS CLI インストールの詳細については AWS CLI、「AWS CLI バージョン 2 のインストール」を参照してください。
- AWS CLI 設定されていることを確認します。設定の詳細については AWS CLI、「AWS CLI バージョン 2 の設定」を参照してください。

- 頻繁に利用される構成設定および認証情報をファイルに保存して AWS CLIによって保守できます。 AWS Ground Station 連絡先を予約および管理するには、これらの設定と認証情報が必要です AWS CLI。設定と認証情報の保存の詳細については、「設定ファイルと認証情報ファイルの設定」を参照してください。

AWS CLI 設定が完了して使用できる状態になったら、AWS Ground Station CLI コマンドリファレンスページを確認して、使用可能なコマンドについて理解してください。 AWS CLI このサービスを使用するときはコマンド構造に従い、groundstation AWS Ground Station 使用するサービスとしてコマンドのプレフィックスを指定してください。 AWS CLI コマンド構造の詳細については、AWS CLI ページの「コマンド構造」を参照してください。コマンド構造の例を以下に示します。

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

衛星に名前を付ける

まず、タグ付けする衛星の ARN を取得する必要があります。これは、AWS CLI のlist-satellites API を介して実行できます。

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

上記の CLI コマンドを実行すると、次のような出力のような出力が返されます。

```
{
    "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"
        }
    ]
}
```

タグ付けする衛星を探して、satelliteArn を書き留めます。タグ付けに関する重要な注意点の 1 つは、tag-resource API にはリージョン ARN が必要であり、list-satellites によって返される ARN はグローバルであるということです。次のステップでは、タグを表示するリージョン (おそらくスケジュールするリージョン) に ARN を拡張する必要があります。この例では、us-west-2 を使用します。この変更により、ARN は以下のように変更されます。変更元:

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

変更先:

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

コンソールにサテライト名を表示するには、キーとなる “Name" があるタグが衛星に必要です。さらに、を使用しているので AWS CLI、引用符はバックスラッシュでエスケープする必要があります。タグは、次のように表示されます。

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

次に、tag-resource API を呼び出して衛星にタグを付けます。これは以下のようにして行うことができます: AWS CLI

```
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\"}
```

これを行うと、 AWS Ground Station サテライトに設定した名前がコンソールに表示されます。

衛星の名前を変更する

衛星の名前を変更する場合は、衛星 ARN を使用して tag-resource をもう一度呼び出すだけで、同じ “Name” キーでタグ内の値が変更されます。これにより既存のタグが更新され、コンソールに新しい名前が表示されます。この呼び出しの例は、次のようになります:

```
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\"}
```

衛星の名前を削除する

衛星に設定された名前は untag-resource API で削除できます。この API では、タグが存在するリージョンの衛星 ARN とタグキーのリストが必要になります。名前の場合、タグのキーは “Name” です。AWS CLI を使用したこの API 呼び出しの例は、次のようになります。

```
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
```

連絡先の予約と管理は AWS CLI

AWS CLI を使用して、 AWS Ground Stationで連絡先を予約および管理できます。 AWS CLI を使用して連絡先を予約および管理する前に、 AWS CLI 次の前提条件を満たす必要があります。

- がインストールされていることを確認してください。 AWS CLI インストールの詳細については AWS CLI、「AWS CLI バージョン 2 のインストール」を参照してください。
- AWS CLI 設定されていることを確認します。設定の詳細については AWS CLI、「AWS CLI バージョン 2 の設定」を参照してください。
- 頻繁に利用される構成設定および認証情報をファイルに保存して AWS CLIによって保守できます。 AWS Ground Station 連絡先を予約および管理するには、これらの設定と認証情報が必要です AWS CLI。設定と認証情報の保存の詳細については、「設定ファイルと認証情報ファイルの設定」を参照してください。

AWS CLI 設定が完了して使用できる状態になったら、AWS Ground Station CLI コマンドリファレンスページを確認して、使用可能なコマンドについて理解してください。 AWS CLI このサービスを使用するときはコマンド構造に従い、groundstation AWS Ground Station 使用するサービスとしてコマンドのプレフィックスを指定してください。 AWS CLI コマンド構造の詳細については、AWS CLI ページの「コマンド構造」を参照してください。コマンド構造の例を以下に示します。

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

以下のトピックを使用して、との連絡先を予約、表示、 AWS CLIキャンセルしてください。

トピック

- 連絡先の表示と一覧表示を行うと AWS CLI
- 連絡先を予約してください AWS CLI

- 連絡先を記述してください AWS CLI
- との連絡を取り消す AWS CLI

連絡先の表示と一覧表示を行うと AWS CLI

CANCELLEDSCHEDULEDまたはとの連絡先を一覧表示するには AWS CLI、aws groundstation list-contacts以下のパラメータを指定して実行します。COMPLETED

- 開始時間 - --start-time <value> でコンタクトの開始時間を指定します。許容される時間値の形式は YYYY-MM-DDTHH:MM:SSZ です。
- 終了時間 - --end-time <value> でコンタクトの終了時間を指定します。許容される時間値の形式は YYYY-MM-DDTHH:MM:SSZ です。
- ステータスリスト - --status-list <value> でコンタクトのステータスを指定します。許容される値は AVAILABLE、CANCELLED、COMPLETED、SCHEDULED などです。すべての有効な値のリストを確認するには、「list-contacts」を参照してください。

AVAILABLE連絡先を一覧表示して表示するには、上記のパラメータに加えて次のパラメータが必要です。 AWS CLI

- 地上局 ID - --ground-station <value> で地上局の ID を指定します。
- ミッションプロファイル ARN - --mission-profile-arn <value> でミッションプロファイルの ARN を指定します。
- 衛星 ARN - --satellite-arn <value> で衛星の ARN を指定します。

list コマンドを使用してリソースを検索できます。パラメータの指定の詳細については、「list-contacts」を参照してください。

使用可能なコンタクトを一覧表示するコマンドの例を以下に示します。

```
aws groundstation --region us-east-2 list-contacts --ground-station 'Ohio 1'
 --mission-profile-arn 'arn:aws:groundstation:us-east-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555' --satellite-arn
 'arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555'
 --start-time '2020-04-10T00:09:22Z' --end-time '2020-04-10T00:11:22' --status-list
 'AVAILABLE'
```

使用可能なコンタクトのリストの例を以下に示します。

```
{
  "contactList": [
    {
      "contactStatus": "AVAILABLE",
      "endTime": "2020-04-15T03:16:35-06:00",
      "groundStation": "Oregon 1",
      "maximumElevation": {
        "unit": "DEGREE_ANGLE",
        "value": 11.22
      },
      "missionProfileArn": "arn:aws:groundstation:us-west-2:111111111111:mission-
profile/11111111-2222-3333-4444-555555555555",
      "region": "us-west-2",
      "satelliteArn":
 "arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555",
      "startTime": "2020-04-15T03:06:08-06:00"
    }
  ]
}
```


連絡先を予約してください AWS CLI

AWS CLI 連絡先を分単位で予約できます。 AWS CLI この機能はに固有のもので、 AWS Ground Station コンソールでは実行できません。

とのコンタクトを予約するには AWS CLI、`aws groundstation reserve-contact`以下のパラメータを指定して実行してください。

- 地上局 ID - `--ground-station <value>` で地上局の ID を指定します。
- ミッションプロファイル ARN - `--mission-profile-arn <value>` でミッションプロファイルの ARN を指定します。
- 衛星 ARN - `--satellite-arn <value>` で衛星の ARN を指定します。
- 開始時間 - `--start-time <value>` でコンタクトの開始時間を指定します。許容される時間値の形式は `YYYY-MM-DDTHH:MM:SSZ` です。
- 終了時間 - `--end-time <value>` でコンタクトの終了時間を指定します。許容される時間値の形式は `YYYY-MM-DDTHH:MM:SSZ` です。

コンタクト予約は非同期プロセスです。`reserve-contact` コマンドへのレスポンスによってコンタクト識別子が示されます。非同期予約プロセスの結果を確認するには、「`describe-contact`」

を使用します。この詳細については、次の「連絡先を記述してください AWS CLI」セクションを参照してください。

list コマンドを使用してリソースを検索できます。パラメータの指定の詳細については、「reserve-contact」を参照してください。

コンタクトを予約するコマンドの例を以下に示します。

```
aws groundstation reserve-contact --ground-station 'Ohio 1' --mission-
profile-arn 'arn:aws:groundstation:us-east-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555' --satellite-arn
 'arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555'
 --start-time '2020-04-10T00:09:22Z' --end-time '2020-04-10T00:11:22'
```

予約が完了したコンタクトの例を以下に示します。

```
{
  "contactId": "11111111-2222-3333-4444-555555555555"
}
```

連絡先を記述してください AWS CLI

との連絡先/予約のステータスを確認するには AWS CLI、CLI コマンドを使用します。describe-contactこれは、非同期コンタクト予約プロセスの結果の検証、進行中のコンタクトのステータスのモニタリング、完了したコンタクトのステータスの判断に役立ちます。

との連絡先を記述するには AWS CLI、aws groundstation describe-contact以下のパラメータを指定して実行します。

- コンタクト ID - --contact-id <value> でコンタクト ID を指定します。

list コマンドを使用してリソースを検索できます。パラメータの指定の詳細については、「describe-contact」を参照してください。

コンタクトを説明するコマンドの例を以下に示します。

```
aws groundstation describe-contact --contact-id 11111111-2222-3333-4444-555555555555
```

スケジュールが完了したコンタクトの例を以下に示します。

```
{
  "groundStation": "Ireland 1",
  "tags": {},
  "missionProfileArn": "arn:aws:groundstation:us-west-2:111111111111:mission-
profile/11111111-2222-3333-4444-555555555555",
  "region": "us-west-2",
  "contactId": "11111111-2222-3333-4444-555555555555",
  "prePassStartTime": 1645850471.0,
  "postPassEndTime": 1645851172.0,
  "startTime": 1645850591.0,
  "maximumElevation": {
      "value": 12.66,
      "unit": "DEGREE_ANGLE"
  },
  "satelliteArn":
 "arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555",
  "endTime": 1645851052.0,
  "contactStatus": "SCHEDULED"
}
```

との連絡を取り消す AWS CLI

との連絡を取り消すには AWS CLI、`aws groundstation cancel-contact`以下のパラメータを指定して実行してください。

- リージョン - `--region <value>` で地上局のリージョンを指定します。
- コンタクト ID - `--contact-id <value>` でコンタクト ID を指定します。

`list` コマンドを使用してリソースを検索できます。パラメータの指定の詳細については、「cancel-contacts」を参照してください

コンタクトを予約するコマンドの例を以下に示します。

```
aws groundstation --region us-east-2 cancel-contact --contact-id
 '11111111-2222-3333-4444-555555555555'
```

キャンセルが完了したコンタクトの例を以下に示します。

```
{
  "contactId": "11111111-2222-3333-4444-555555555555"
```

```
}
```

Amazon EC2 へのデータ配信

AWS Ground Station 連絡先データをアカウントの Amazon Simple Storage Service (Amazon S3) バケットに非同期的に配信するか、アカウントの Amazon Elastic Compute Cloud (Amazon EC2) インスタンスとの間で同期的にストリーミングします。次の手順では、Amazon EC2 インスタンスとの間で連絡先データをストリーミングするために必要なリソースを設定する方法について説明します。Amazon S3 へのデータ配信の詳細については、の開始方法 AWS Ground Stationガイドを参照してください。

トピック

ステップ 1: EC2 SSH キーペアを作成する

まだキーペアをお持ちでない場合は、 AWS データを受け取る予定のリージョンごとに Amazon EC2 コンソールで新しいキーペアを作成します。以下の手順に従います。

1. で AWS Management Console、 AWS 連絡先を予約する予定のリージョンを選択します。 AWS 選択したすべてのリージョンにkey pair を作成する必要があります。

   > **Note**
   >
   > AWS Ground Station まだ一部の地域ではご利用いただけません。 AWS Ground Station AWS ご希望の地域でサポートされていることを確認してください。 AWS Ground Station アンテナの場所の詳細については、「AWS Ground Station に関する FAQ」を参照してください。

2. 「Amazon EC2 ユーザーガイド」の「キーペアの作成」ガイドに従ってキーペアを作成します。
3. 必要に応じて、 AWS 他のリージョンについても同じ手順を繰り返します。

ステップ 2: VPC を設定する

VPC の完全な設定は、このガイドでは扱いません。カスタマイズ済みの既存の VPC がない場合は、AWS アカウントに作成されているデフォルトの VPC を使用できます。パブリック IP アドレスをアタッチせずに Amazon EC2 インスタンスに SSH 接続できるように、Linux 踏み台を VPC に追加することをお勧めします。VPC での Linux 踏み台の設定の詳細については、「AWS での Linux 踏み台ホスト」を参照してください。

参考までに、Linux AWS 環境に要塞ホストをすばやく追加する手順を以下に示します。これは必須ではありませんが、推奨されるベストプラクティスです。

1. AWS アカウントにログインします。
2. 「AWS クラウド上の Linux 踏み台ホスト: クイックスタートリファレンスデプロイ」ページで、[Launch Quick Start (for new VPC) (クイックスタートを起動 (新しい VPC の場合))] を選択します。
3. [スタックの作成] ページで、[次へ] を選択します。テンプレートはあらかじめ入力されています。
4. [スタックの指定] 詳細ページで、次のボックスを編集および変更します。

    a. [スタックの名前] ボックスにホストのスタック名を入力します。

    b. [Availability Zones (アベイラビリティーゾーン)] で、VPC 内のサブネットに使用するアベイラビリティーゾーンを選択します。少なくとも 2 つのアベイラビリティーゾーンを選択する必要があります。

    c. [Allowed bastion external access CIDR (許可された踏み台外部アクセス CIDR)] に、SSH アクセスを有効にする CIDR ブロックを入力します。不明な場合は、0.0.0.0/0 の値を使用して、SSH キーを持つ任意のホストからの SSH アクセスを有効にすることができます。

    d. [Key pair name (キーペア名)] で、「the section called “ステップ 1: EC2 SSH キーペアを作成する”」で作成したキーペア名を選択します。

    e. [Bastion instance type] で、[t2.micro] を選択します。

    > **Important**
    >
    > t2.micro インスタンスタイプは、欧州 (ストックホルム) リージョン (eu-north-1) では使用できません。ヨーロッパ (ストックホルム) リージョン (eu-north-1) AWS Ground Station で使用している場合は、t3.micro を選択してください。

    f. TCP 転送の場合は、true を選択します。

g. (オプション) 必要に応じて他の編集や変更を行います。デプロイのカスタマイズでは、VPC 構成の変更、踏み台ホストインスタンスの数と種類の選択、TCP または X11 フォワーディングの有効化、踏み台ホストの既定またはカスタムのバナーの有効化を行うことができます。

h. [次へ] をクリックします。

5. [スタックオプションの設定] ページで、必要に応じて変更または編集を行います。
6. [次へ] をクリックします。
7. 踏み台ホストの詳細を確認し、2 つの [機能] 確認を選択します。[スタックの作成] を選択します。

ステップ 3: テンプレートを選択してカスタマイズする AWS CloudFormation

現在、Contact ごとに VPC への複数のデータストリームを設定できます。これらのデータストリームは、2 つの異なる形式で使用できます。VITA-49 Signal/IP データを含むデータストリームは、最大 54 MHz の帯域幅の S バンド信号および X バンド信号に対して設定できます。VITA-49 拡張データ/IP は、最大 500 MHz の帯域幅の復調および/またはデコードされた X バンド信号に対して設定できます。

衛星をオンボーディングしたら、ミッションプロファイルを定義し、衛星との間でデータストリームを処理またはプッシュするインスタンスを作成する必要があります。このプロセスを支援するために、 AWS CloudFormation 公共放送衛星を使用する設定済みのテンプレートを用意しています。これらのテンプレートを使うと、簡単に使い始めることができます。 AWS Ground Station詳細については AWS CloudFormation、「AWS とは CloudFormation?」を参照してください。

重要な点として、Amazon EC2 インスタンスの Data Defender の localhost 側をリッスンするデータ処理ソフトウェアまたはデータストレージソフトウェアが必要であることに注意してください。このソフトウェアは、コンタクト中に Amazon EC2 インスタンスに配信されたデータを保存または処理するために使用します。

Amazon EC2 インスタンス設定を構成する

AWS CloudFormation このセクションで提供されるテンプレートは、デフォルトで Amazon EC2 m5.4xlarge インスタンスタイプを使用するように設定されています。ただし、ユースケースに合わせて Amazon EC2 インスタンス設定をカスタマイズおよび選択することをお勧めします。インスタ

ンス設定を選択する際には、ストレージ I/O や CPU パフォーマンスなどの要件を考慮する必要があります。たとえば、レシーバーインスタンスでソフトウェアモデムを実行する場合、より多くのコアとより高いクロック速度を備えたコンピューティング最適化インスタンスが必要になることがあります。ユースケースに適したインスタンス設定を判断する最善の方法は、ワークロードでインスタンス設定をテストすることです。Amazon EC2 では、インスタンス設定を簡単に切り替えることができます。テンプレートを使用し、必要に応じてインスタンス設定をカスタマイズします。

一般的な推奨事項として、アップリンクとダウンリンクの拡張ネットワーキングをサポートするインスタンス (AWS Nitro System など) AWS Ground Station の使用を推奨しています。拡張ネットワーキングの詳細については、「Linux インスタンスで Elastic Network Adapter (ENA) を使用して拡張ネットワーキングを有効にする」を参照してください。

Amazon EC2 インスタンスタイプの設定に加えて、 AWS CloudFormation テンプレートはインスタンスに使用するベースの Amazon マシンイメージ (AMI) を設定します。 AWS Ground Station ベースには、EC2 インスタンスにプリインストールされているサービスからデータを受信するために必要なソフトウェアが含まれています。 AMI の詳細については、「Amazon マシンイメージ (AMI)」を参照してください。

手動でリソースを作成および構成する

AWS CloudFormation このセクションのサンプルテンプレートは、衛星通信の実行を開始するのに必要なすべてのリソースを設定します。衛星通信の実行を開始するために必要なリソースを手動で作成して構成する場合は、次の操作を行う必要があります。

- AWS Ground Station コンフィグを作成します。設定を手動で作成する方法の詳細については、「Create AWS Ground Station Config AWS CLI コマンドリファレンス」または「Create Config API リファレンス」を参照してください。
- AWS Ground Station ミッションプロファイルを作成します。ミッションプロファイルを手動で作成する方法の詳細については、「 AWS Ground Station ミッションプロファイルの作成 AWS CLI コマンドリファレンス」または「ミッションプロファイルの作成 API リファレンス」を参照してください。
- AWS Ground Station データフローエンドポイントグループを作成します。データフローエンドポイントグループを手動で作成する方法の詳細については、「 AWS Ground Station データフローエンドポイントグループの作成 AWS CLI コマンドリファレンス」または「データフローエンドポイントグループの作成 API リファレンス」を参照してください。
- EC2 インスタンスを作成します。で使用する EC2 インスタンスを手動で作成する方法の詳細については、を参照してください。 AWS Ground StationAmazon EC2 インスタンスを作成する

- EC2 AWS Ground Station インスタンスとの間でデータを送受信できるように EC2 インスタンスのセキュリティグループを設定します。手動での EC2 インスタンスのセキュリティグループ設定の設定の詳細については、「create-security-group AWS CLI コマンドリファレンス」または「CreateSecurityGroup API リファレンス」を参照してください。

テンプレートの選択

AWS Ground Station には、サービスの使用方法を示すテンプレートが用意されており、さまざまな方法でアクセスできます。このガイドを使用して、適切なテンプレートを見つけてください。

事前設定されたテンプレートの使用

事前設定されたテンプレートを使用して、Aqua、SNPP、JPSS-1/NOAA-20、および Terra 衛星からダイレクトブロードキャストデータを受信できます。これらのテンプレートには、問い合わせのスケジュールと実行に必要なAWS CloudFormation リソースが含まれています。AquaSnppJpssテンプレートには、 AWS CloudFormation 復調およびデコードされたダイレクトブロードキャストデータを受信するのに必要なリソースが含まれています。NASA Direct Readout Labs ソフトウェア (RT-STPS および IPOPP) を使用してデータを処理する場合は、このテンプレートを開始点として使用してください。この AquaSnppJpssTerraDigIF テンプレートは、生のデジタル化された中間周波数 (DigIF) ダイレクトブロードキャストデータを受信するために必要な AWS CloudFormation リソースで構成されます。このテンプレートは、ソフトウェア定義無線 (SDR) を使用してデータを処理するための出発点として使用します。DirectBroadcastSatelliteWbDigIfEc2DataDeliveryこのテンプレートには、未加工のワイドバンドデジタル中間周波数 (DigIF) ダイレクトブロードキャストデータを Agent AWS CloudFormation 経由で受信するのに必要なリソースが含まれています。AWS Ground Station

ナローバンドデータ配信テンプレート:

- the section called “AquaSnppJpss テンプレート (ナローバンド)”
- the section called “AquaSnppJpssTerraDigIF テンプレート (ナローバンド)”

ブロードバンド DigIF データ配信テンプレート:

- the section called “ダイレクトブロードキャスト衛星ブロードバンド DigIF テンプレート (ワイドバンド)”

> **Important**
>
> テンプレートを使用して AMI にアクセスするには、衛星をサービスにオンボーディングする必要があります。 AWS CloudFormation

独自の衛星の使用

独自の衛星を設定するには、異なるパラメータとリソースのセットが必要です。この作業をお客様自身で行うのは困難です。 AWS Ground Station このチームは、お客様独自のサテライトを使用するよう設定したり、ダウンリンク、アップリンク、アップリンクエコーストリームのリソースを設定したりするお手伝いをします。独自のサテライトをで使用するように設定するには AWS Ground Station、AWS Support にお問い合わせください。

テンプレートへのアクセス

テンプレートは、以下のリージョンの Amazon S3 バケットでアクセスできます。次のリンクでは、リージョン S3 エンドポイントを使用しています。<us-west-2> AWS CloudFormation スタックを作成しているリージョンに変更します。

```
s3://groundstation-cloudformation-templates-us-west-2/
```

AWS CLIを使用してテンプレートをダウンロードすることもできます。の設定方法については AWS CLI、「の設定」を参照してください AWS CLI。

AquaSnppJpss テンプレート (ナローバンド)

AWS CloudFormation AquaSnppJpss.ymlという名前のテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20 衛星のデータ受信をすばやく開始できるように設計されています。Amazon EC2 インスタンスと、コンタクトをスケジュールしたり、 AWS Ground Station 復調/デコードされたダイレクトブロードキャストデータを受信したりするのに必要なリソースが含まれています。このテンプレートは、NASA Direct Readout Labs ソフトウェア (RT-STPS および IPOPP) を使用してデータを処理する予定の場合に適しています。

Aqua、SNPP、および JPSS-1/NOAA-20 がアカウントにオンボーディングされていない場合は、「カスタマーオンボーディング」を参照してください。

> **Important**
>
> テンプレートを適用する前に Amazon EC2 インスタンスを停止する必要があります。使用する準備ができるまでインスタンスが停止していることを確認します。

テンプレートにアクセスするには、カスタマーのオンボーディング S3 バケットにアクセスします。以下のリンクでは、リージョン S3 バケットを使用していることに注意してください。<us-west-2>スタックを作成しているリージョンに変更します。 AWS CloudFormation

> **Note**
>
> 以下の手順は、YAML を使用します。ただし、テンプレートは YAML 形式と JSON 形式の両方で使用できます。JSON を使用するには、<.yml> を <.json> に置き換えます。

を使用してテンプレートをダウンロードするには AWS CLI、以下のコマンドを使用します。

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml .
```

ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml
```

AWS CloudFormation 次のリンクを使用してテンプレートを直接指定できます。

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/AquaSnppJpss.yml
```

テンプレートでどのようなリソースを定義しますか?

AquaSnppJpss テンプレートは以下のリソースが含まれます。

- データ配信サービスの役割- AWS Ground Station データをストリーミングするために、アカウント内の ENI を作成または削除する役割を引き受けます。
- (オプション) レシーバーインスタンス-を使用して衛星との間でデータを送受信する Amazon EC2 インスタンス。 AWS Ground Station

  - インスタンスセキュリティグループ - Amazon EC2 インスタンスのセキュリティグループ。
  - インスタンスロール - Amazon EC2 インスタンスのロール。
  - インスタンスプロファイル - Amazon EC2 インスタンスのインスタンスプロファイル。
  - クラスタープレイスメントグループ - Amazon EC2 インスタンスが起動するプレイスメントグループ。
- Dataflow エンドポイントセキュリティグループ-によって作成された elastic network interface AWS Ground Station が属するセキュリティグループ。デフォルトでは、このセキュリティグループは VPC 内の任意の IP AWS Ground Station アドレスにトラフィックをストリーミングすることを許可します。これは、特定の IP アドレスのセットにトラフィックを制限する方法で変更できます。
- レシーバーインスタンスネットワークインターフェイス- AWS Ground Station 接続用の固定 IP アドレスを提供する伸縮自在なネットワークインターフェイス。これは、`eth1` のレシーバーインスタンスにアタッチされます。
- レシーバーインスタンスインターフェイスのアタッチ - Amazon EC2 インスタンスにアタッチする Elastic Network Interface。
- (オプション) CloudWatch イベントトリガー- AWS Lambda CloudWatch AWS Ground Station コンタクトの前後に送信されるイベントを使用してトリガーされる関数。 AWS Lambda この関数は Receiverインスタンスを起動し、必要に応じて停止します。
- (オプション) コンタクトの EC2 検証 - Lambda を使用して SNS 通知でコンタクトに Amazon EC2 インスタンスの検証システムをセットアップするオプション。現在の使用状況によっては、料金が発生する可能性があることに注意してください。
- データフローエンドポイントグループ-Satellite AWS Ground Station との間でデータを送受信するために使用するエンドポイントを定義するデータフローエンドポイントグループ。データフローエンドポイントグループの作成の一環として、データをストリーミングするための Elastic Network Interface AWS Ground Station をアカウントに作成します。
- 追跡Config- AWS Ground Station 追跡設定は、衛星が空を移動するときにアンテナシステムがどのように追跡するかを定義します。
- Ground Station Amazon マシンイメージ取得 Lambda - インスタンスにインストールされているソフトウェアと任意の AMI を選択するオプション。ソフトウェアのオプションは、`DDX 2.6.2 Only` と `DDX 2.6.2 with qRadio 3.6.0` です。ワイドバンド DigIF データ配信と AWS Ground Station Agent を使用する場合は、を使用してください。AquaSnppJpssTerraDigIF テンプレート (ナローバンド)これらのオプションは、追加のソフトウェア更新プログラムおよび機能がリリースされるにつれて引き続き拡張されます。

さらに、このテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20 衛星のために次のリソースを提供します。

- JPSS-1/NOAA-20 と SNPP 用のダウンリンク復調/復号設定、および Aqua 用のダウンリンク復調/復号設定
- JPSS-1/NOAA-20 と SNPP 用のミッションプロファイル、および Aqua 用のミッションプロファイル

このテンプレートでは衛星の値とパラメータが入力済みです。これらのパラメータにより、 AWS Ground Station これらのサテライトをすぐに使用することが容易になります。 AWS Ground Station このテンプレートを使用する際には、独自の値を設定する必要はありません。ただし、値をカスタマイズして、ユースケースに合わせてテンプレートを使用することもできます。

データはどこで受信できますか?

データフローエンドポイントグループは、テンプレートで作成されるレシーバーインスタンスのネットワークインターフェイスを使用するように設定されます。レシーバーインスタンスは Data Defender を使用して、 AWS Ground Station データフローエンドポイントによって定義されたポートからデータストリームを受信します。受信すると、受信側インスタンスのループバックアダプターの UDP ポート 50000 を介してデータを消費できるようになります。データフローエンドポイントグループの設定について詳しくは、「グループ」を参照してください。AWS::GroundStation::DataflowEndpoint

AquaSnppJpssTerraDigIF テンプレート (ナローバンド)

AWS CloudFormation `AquaSnppJpssTerraDigIF.yml`という名前のテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20、Terra衛星のデジタル化された中間周波数 (DigIF) データの受信をすばやく開始できるように設計されています。Amazon EC2 インスタンスと、未加工の DigIF AWS CloudFormation ダイレクトブロードキャストデータを受信するために必要なリソースが含まれています。このテンプレートは、ソフトウェア定義無線 (SDR) を使用してデータを処理するための出発点として適しています。

Aqua、SNPP、JPSS-1/NOAA-20、および Terra がアカウントにオンボーディングされていない場合は、「カスタマーオンボーディング」を参照してください。

> **Important**
>
> テンプレートを適用する前に Amazon EC2 インスタンスを停止する必要があります。使用する準備ができるまでインスタンスが停止していることを確認します。

テンプレートにアクセスするには、カスタマーのオンボーディング S3 バケットにアクセスします。以下のリンクでは、リージョン S3 バケットを使用していることに注意してください。<us-west-2> AWS CloudFormation スタックを作成しているリージョンに変更します。

> **Note**
>
> 以下の手順は、YAML を使用します。ただし、テンプレートは YAML 形式と JSON 形式の両方で使用できます。JSON を使用するには、<.yml> を <.json> に置き換えます。

を使用してテンプレートをダウンロードするには AWS CLI、以下のコマンドを使用します。

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
AquaSnppJpssTerraDigIF.yml .
```

ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-
west-2/AquaSnppJpssTerraDigIF.yml
```

AWS CloudFormation 次のリンクを使用してテンプレートを直接指定できます。

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
AquaSnppJpssTerraDigIF.yml
```

テンプレートでどのようなリソースを定義しますか?

AquaSnppJpssTerraDigIF テンプレートは以下のリソースが含まれます。

- データ配信サービスの役割- AWS Ground Station データをストリーミングするために、アカウント内の ENI を作成または削除する役割を引き受けます。

- (オプション) レシーバーインスタンス-を使用して衛星との間でデータを送受信する Amazon EC2 インスタンス。 AWS Ground Station
  - インスタンスセキュリティグループ - Amazon EC2 インスタンスのセキュリティグループ。
  - インスタンスロール - Amazon EC2 インスタンスのロール。
  - インスタンスプロファイル - Amazon EC2 インスタンスのインスタンスプロファイル。
  - クラスタープレイスメントグループ - Amazon EC2 インスタンスが起動するプレイスメントグループ。
- Dataflow エンドポイントセキュリティグループ-によって作成された elastic network interface AWS Ground Station が属するセキュリティグループ。デフォルトでは、このセキュリティグループは VPC 内の任意の IP AWS Ground Station アドレスにトラフィックをストリーミングすることを許可します。これは、特定の IP アドレスのセットにトラフィックを制限する方法で変更できます。
- レシーバーインスタンスネットワークインターフェイス- AWS Ground Station 接続用の固定 IP アドレスを提供する伸縮自在なネットワークインターフェイス。これは、`eth1` のレシーバーインスタンスにアタッチされます。
- レシーバーインスタンスインターフェイスのアタッチ - Amazon EC2 インスタンスにアタッチする Elastic Network Interface。
- (オプション) CloudWatch イベントトリガー- AWS Lambda CloudWatch AWS Ground Station コンタクトの前後に送信されるイベントを使用してトリガーされる関数。 AWS Lambda この関数は Receiverインスタンスを起動し、必要に応じて停止します。
- (オプション) コンタクトの EC2 検証 - Lambda を使用して SNS 通知でコンタクトに Amazon EC2 インスタンスの検証システムをセットアップするオプション。現在の使用状況によっては、料金が発生する可能性があることに注意してください。
- データフローエンドポイントグループ-Satellite AWS Ground Station との間でデータを送受信するために使用するエンドポイントを定義するデータフローエンドポイントグループ。データフローエンドポイントグループの作成の一環として、データをストリーミングするための Elastic Network Interface AWS Ground Station をアカウントに作成します。
- 追跡Config- AWS Ground Station 追跡設定は、衛星が空を移動するときにアンテナシステムがどのように追跡するかを定義します。
- ダウンリンク Dig IF エンドポイントの設定 - 衛星からデータをダウンリンクするために使用される定義されたエンドポイント。
- Ground Station Amazon マシンイメージ取得 Lambda - インスタンスにインストールされているソフトウェアと任意の AMI を選択するオプション。ソフトウェアのオプションは、`DDX 2.6.2`

`Only` と `DDX 2.6.2 with qRadio 3.6.0` です。これらのオプションは、追加のソフトウェア更新プログラムおよび機能がリリースされるにつれて引き続き拡張されます。

さらに、このテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20、および Terra 衛星のために次のリソースを提供します。

- Aqua、SNPP、JPSS-1/NOAA-20、および Terra のためのダウンリンク DigIF アンテナ設定。
- JPSS-1/NOAA-20 と SNPP のためのミッションプロファイル、Aqua のためのミッションプロファイル、および Terra のためのミッションプロファイル。

このテンプレートでは衛星の値とパラメータが入力済みです。これらのパラメータにより、 AWS Ground Station これらの衛星をすぐに使用することが容易になります。 AWS Ground Station このテンプレートを使用する際には、独自の値を設定する必要はありません。ただし、値をカスタマイズして、ユースケースに合わせてテンプレートを使用することもできます。

データはどこで受信できますか?

データフローエンドポイントグループは、テンプレートで作成されるレシーバーインスタンスのネットワークインターフェイスを使用するように設定されます。レシーバーインスタンスは Data Defender を使用して、 AWS Ground Station データフローエンドポイントによって定義されたポートからデータストリームを受信します。受信すると、受信側インスタンスのループバックアダプターの UDP ポート 50000 を介してデータを消費できるようになります。データフローエンドポイントグループの設定について詳しくは、「グループ」を参照してください。AWS::GroundStation::DataflowEndpoint

ダイレクトブロードキャスト衛星ブロードバンド DigIF テンプレート (ワイドバンド)

AWS CloudFormation `DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml`という名前のテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20、Terra衛星のデジタル化された中間周波数（DigIF）データの受信をすばやく開始できるように設計されています。Amazon EC2 インスタンスと、未加工の DigIF AWS CloudFormation ダイレクトブロードキャストデータを受信するために必要なリソースが含まれています。このテンプレートは、ソフトウェア定義無線 (SDR) を使用してデータを処理するための出発点として適しています。

Aqua、SNPP、JPSS-1/NOAA-20、および Terra がアカウントにオンボーディングされていない場合は、「カスタマーオンボーディング」を参照してください。

> **Important**
>
> テンプレートを適用する前に Amazon EC2 インスタンスを停止する必要があります。使用する準備ができるまでインスタンスが停止していることを確認します。

テンプレートにアクセスするには、カスタマーのオンボーディング S3 バケットにアクセスします。以下のリンクでは、リージョン S3 バケットを使用していることに注意してください。<us-west-2> AWS CloudFormation スタックを作成しているリージョンに変更します。

> **Note**
>
> 以下の手順は、YAML を使用します。ただし、テンプレートは YAML 形式と JSON 形式の両方で使用できます。JSON を使用するには、<.yml> を <.json> に置き換えます。

を使用してテンプレートをダウンロードするには AWS CLI、以下のコマンドを使用します。

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/agent/ec2_delivery/
DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml .
```

ブラウザで以下の URL に移動して、テンプレートをコンソールで表示およびダウンロードできます。

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-
west-2/agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

AWS CloudFormation 次のリンクを使用してテンプレートを直接指定できます。

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/
agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

テンプレートでどのようなリソースを定義しますか?

`DirectBroadcastSatelliteWbDigIfEc2DataDelivery` テンプレートは以下のリソースが含まれます。

- (オプション) レシーバーインスタンス-を使用して衛星との間でデータを送受信する Amazon EC2 インスタンス。 AWS Ground Station

  - インスタンスセキュリティグループ - Amazon EC2 インスタンスのセキュリティグループ。
  - インスタンスロール - Amazon EC2 インスタンスのロール。
  - インスタンスプロファイル - Amazon EC2 インスタンスのインスタンスプロファイル。
  - クラスタープレイスメントグループ - Amazon EC2 インスタンスが起動するプレイスメントグループ。
- データ配信キー- AWS KMS データフローの暗号化に使用されるキー。
- Ground Station キーロール- AWS KMS データフローを復号化するためにキーにアクセスして使用することを想定する IAM ロール AWS Ground Station
- Ground Station キーアクセスポリシー- AWS Ground Station データ配信キーに対して実行できるアクションを定義する IAM ポリシー
- レシーバーインスタンスの elastic network interface-(条件付き) Elastic Network Interface は、PublicSubnetIdif を指定すると指定されたサブネットに作成されます。これは、レシーバーインスタンスがプライベートサブネットにある場合に必要です。elastic network interface は EIP に関連付けられ、レシーバーインスタンスにアタッチされます。
- レシーバーインスタンス Elastic IP- AWS Ground Station 接続先となるエラスティック IP。これはレシーバーインスタンスまたは elastic network interface にアタッチされます。
- 以下のいずれかの Elastic IP アソシエーション:
  - レシーバーインスタンスと Elastic IP アソシエーション-Elastic IP とレシーバーインスタンスの関連付け (PublicSubnetId指定されていない場合)。これには、SubnetIdパブリックサブネットを参照する必要があります。
  - レシーバーインスタンスの elastic network interface to Elastic IP アソシエーション-Elastic IP とレシーバーインスタンスのエラスティックネットワークインターフェイス (PublicSubnetId指定されている場合) の関連付け。
- (オプション) CloudWatch イベントトリガー- AWS Lambda CloudWatch AWS Ground Station コンタクトの前後に送信されるイベントを使用してトリガーされる関数。 AWS Lambda この関数は Receiverインスタンスを起動し、必要に応じて停止します。
- (オプション) コンタクトの EC2 検証 - Lambda を使用して SNS 通知でコンタクトに Amazon EC2 インスタンスの検証システムをセットアップするオプション。現在の使用状況によっては、料金が発生する可能性があることに注意してください。
- データフローエンドポイントグループ-Satellite AWS Ground Station [との間でデータを送受信するために使用するエンドポイントを定義するデータフローエンドポイントグループ](#)。
- 追跡Config- AWS Ground Station [追跡設定は](#)、衛星が空を移動するときにアンテナシステムがどのように追跡するかを定義します。

さらに、このテンプレートは、Aqua、SNPP、JPSS-1/NOAA-20、および Terra 衛星のために次のリソースを提供します。

- JPSS-1/NOAA-20 と SNPP のダウンリンク設定、Aqua のダウンリンク設定、Terra のダウンリンク設定。
- JPSS-1/NOAA-20 と SNPP のためのミッションプロファイル、Aqua のためのミッションプロファイル、および Terra のためのミッションプロファイル。

このテンプレートでは衛星の値とパラメータが入力済みです。これらのパラメータにより、 AWS Ground Station これらの衛星をすぐに使用することが容易になります。 AWS Ground Station このテンプレートを使用する際には、独自の値を設定する必要はありません。ただし、値をカスタマイズして、ユースケースに合わせてテンプレートを使用することもできます。

データはどこで受信できますか?

データフローエンドポイントグループは、テンプレートで作成されるレシーバーインスタンスのネットワークインターフェイスを使用するように設定されます。レシーバーインスタンスは AWS Ground Station Agent を使用して、 AWS Ground Station データフローエンドポイントによって定義されたポートからデータストリームを受信します。データフローエンドポイントグループの設定の詳細については、「グループ」を参照してください。AWS::GroundStation::DataflowEndpoint AWS Ground Station Agent の詳細については、を参照してください。AWS Ground Station エージェントユーザーガイド

Amazon EC2 インスタンスを作成する

> **Note**
>
> リソース AWS Ground Station (Amazon EC2 インスタンスを含む) を手動で作成する必要も推奨もされていません。 AWS Ground Station AWS CloudFormation そのための既成のテンプレートが提供されています (ステップ 3: テンプレートを選択してカスタマイズする AWS CloudFormation詳細については「」を参照)。 AWS CloudFormation テンプレートの使用が自分のユースケースに合っていない場合は、読み続けてください。

AWS Ground Station には、Amazon EC2 インスタンスでナローバンドまたはワイドバンドのデータ配信を行うために必要なソフトウェアがプリロードされている Amazon EC2 AMI が用意されています。 DigIf

> **Important**
>
> AMI にアクセスするには、サテライトをサービスにオンボーディングする必要があります。 AWS Ground Station

Amazon EC2 AMI と DataDefender

この AMI DataDefender にはソフトウェアがあらかじめインストールされており、ナローバンドデータ配信のダウンリンク連絡先に使用されます。

この AMI の命名規則 は `groundstation-al2-ddx$DDX_VERSION-ami-$DATE_PUBLISHED` です。新しい DDX AMI は、新しい AL2 Amazon EC2 AMI が公開された直後に公開されます。 AWS Ground Station DataDefender がソフトウェアの新しいバージョンをサポートすることを決定した場合、更新されたバージョンを使用して新しい AMI が公開されます。

で AWS Ground Station AMI を選択する DataDefender

AWS Ground Station AMI には、Amazon EC2 コンソールの [AMI] タブからアクセスできます。そのページに移動すると、[プライベートイメージ] フィルタで AMI にアクセスできるようになります。

AMI を公開日でソートし、最近公開された `groundstation-al2-ddx$DDX_VERSION-ami-$DATE_PUBLISHED` という名前の AMI を使用することをお勧めします。

Amazon EC2 AMI AWS Ground Station とエージェント

この AMI は AWS Ground Station Agent にあらかじめインストールされており、ワイドバンド DigIF ダウンリンクコンタクトに使用されます。

この AMI の命名規則は、`groundstation-al2-gs-agent-ami-*` です。* は AMI が構築された日付です。 AWS Ground Station 新しいエージェント AMI は、新しい AL2 Amazon EC2 AMI が公開された直後、 AWS Ground Station またはエージェント RPM の新しいバージョンがリリースされた直後に公開されます。

AWS Ground Station エージェントの詳細については、「」を参照してください。AWS Ground Station エージェントユーザーガイド

AWS Ground Station エージェント AMI の選択

AWS Ground Station エージェント AMI には、Amazon EC2 コンソールの [AMI] タブからアクセスできます。そのページに移動すると、[パブリックイメージ] フィルタで AMI にアクセスできるようになります。

AMI を公開日でソートし、最近公開された `groundstation-al2-gs-agent-ami-$DATE_PUBLISHED` という名前の AMI を使用することをお勧めします。

ステップ 4: スタックを設定する AWS CloudFormation

ユースケースに最適なテンプレートを選択したら、 AWS CloudFormation スタックを設定します。この手順で作成されるリソースは、作成時のリージョンに設定されます。これには、データの配信先リージョンを決定するミッションプロファイルとそのプロパティが含まれます。

1. で AWS Management Console、[サービス] > を選択します CloudFormation。
2. ナビゲーションペインで、[Stacks] を選択します。次に、[スタックの作成] - [With new resources (standard)] の順に選択します。
3. [スタックの作成] ページで、次のいずれかを実行して、「the section called “テンプレートの選択”」で選択したテンプレートを指定します。

   a. テンプレートソースとして [Amazon S3 URL] を選択し、Amazon S3 URL で使用するテンプレートの URL をコピーして貼り付けます。[次へ] を選択します。

   b. テンプレートソースとして [テンプレートファイルをアップロード] を選択し、[ファイルを選択] を選択します。「the section called “テンプレートの選択”」でダウンロードしたテンプレートをアップロードします。[次へ] を選択します。
4. [スタックの指定] 詳細ページで、次の変更を行います。

   a. [スタックの名前] ボックスに名前を入力します。将来のエラーの可能性を減らすために、単純な名前を使用することをお勧めします。

   b. CloudWatchEventActionsでは、 CloudWatch コンタクトの前後のイベントトリガーに対して実行するアクションを選択します。

   c. CreateEC2 VerificationForContacts では、SNS 通知付きの連絡先用に EC2 インスタンスの検証システム (Lambda を使用) を設定するかどうかを選択します。現在の使用状況によっては、料金が発生する可能性があることに注意してください。

   d. CreateReceiverInstanceでは、Amazon EC2 レシーバーインスタンスを作成するかどうかを選択します。

e. 「the section called “ステップ 1: EC2 SSH キーペアを作成する”」で作成した SSH キーを選択します。

f. Amazon EC2 インスタンスを作成したい場所を選択します。SubnetId

AWS Ground Station Agent を使用する場合、インスタンスまたは elastic network interface の配置にパブリックサブネットが必要です。インスタンスを配置するプライベートサブネットを指定する場合は、 AWS Ground Station エージェントで使用するパブリックサブネット PublicSubnetId(以下を参照) も指定する必要があります。SubnetId

エージェント以外のユースケースでは、Amazon EC2 インスタンスをプライベートサブネットに配置することをベストプラクティスとしてお勧めしますが、必須ではありません。「AWS クラウド上の Linux 踏み台ホスト: クイックスタートリファレンスデプロイ」を使用して、the section called “ステップ 2: VPC を設定する” でアカウントをすでにプライベートサブネットで設定していない場合は、プライベートサブネットを自動的に作成できます。

> **Note**
>
> 組織によっては、Amazon EC2 インスタンス専用の別のサブネットを持っている場合があります。

g. (オプション) プライベートサブネット内のインスタンスで AWS Ground Station Agent PublicSubnetIdを使用する場合にのみ使用するように選択してください。でプライベートサブネットを指定した場合は必須ですSubnetId。

このサブネットは、アカウント内のので指定されたものと同じアベイラビリティーゾーンにある必要がありますSubnetId。PublicSubnetIdを指定すると、指定したパブリックサブネットに Elastic Network Interface が作成され、インスタンスにアタッチされます。このインターフェースは、 AWS Ground Station で指定されたプライベートサブネットに配置されたインスタンスからのエージェントネットワークアクセスに使用されますSubnetId。

h. 「the section called “ステップ 2: VPC を設定する”」で作成した VPC スタックを選択します。

i. [次へ] をクリックします。

5. Amazon EC2 インスタンスのスタックオプションと詳細オプションを設定します。

a. [タグ] セクションと [アクセス権限] セクションでタグとアクセス権限を追加します。

b. [スタックポリシー]、[ロールバック設定]、[通知オプション]、および [スタック作成オプション] を変更します。

c. [次へ] をクリックします。

6. スタックの詳細を確認したら、[CAPABILITY] 確認を選択し、[スタックの作成] を選択します。

ステップ 5: FE プロセッサ/無線をインストールして設定する

AWS CloudFormation テンプレートで定義されている Amazon EC2 インスタンスには、フロントエンド (FE) プロセッサまたはソフトウェア定義無線 (SDR) がデフォルトでインストールされていません。 AWS Ground Station アンテナシステムとの間でストリーミングされる VITA-49 パケットを処理するには、FE プロセッサまたは SDR をインストールする必要があります。

FE プロセッサまたは SDR をインストールして設定する方法は、どの FE プロセッサまたは SDR を使用するかによって異なります。FE プロセッサまたは SDR のインストールについては、このユーザーガイドで扱いません。

FE プロセッサー/無線をインストールして設定するには、AWS サポートにお問い合わせください。

> **Important**
>
> Data Defender との間の DTLS データストリームの利点を活用するには、 AWS CloudFormation テンプレートによって作成されたインスタンスで FE プロセッサまたは SDR を実行するのがベストプラクティスです。

次のステップ

AWS Ground Station アカウントとリソースの設定が完了し、使用できる状態になりました。 AWS Ground Station これらのリソースはコンソールで使用できるようになり、選択した衛星の衛星データの入力、アンテナ位置の特定、通信、アンテナ時間のスケジュール設定を行うことができます。また、さまざまなツールを使用してアクティビティをモニタリングし、アラームを設定することもできます。

詳細については、以下のトピックを参照してください。

- コンタクトの一覧表示と予約
- モニタリング AWS Ground Station

クロスリージョンのデータ配信サービスの使用

AWS Ground Station クロスリージョンデータ配信機能を使用すると、アンテナから AWS リージョンの Amazon EC2 インスタンスにデータを柔軟に送信できます。現在、リージョン間のデータ配信は、Amazon S3 バケットで問い合わせデータを受信するときに、 AWS Ground Station サポートされているすべてのリージョンで利用できます。Amazon EC2 へのデータ配信を利用する場合、次の antenna-to-destination リージョンでのみ使用できます。

- 米国東部 (オハイオ) リージョン (us-east-2) から米国西部 (オレゴン) リージョン (us-west-2)
- 米国西部 (オレゴン) リージョン (us-west-2) から米国東部 (オハイオ) リージョン (us-east-2)

クロスリージョンデータ配信を使用するには、 AWS CloudFormation テンプレートを設定する必要があります。 AWS CloudFormation テンプレートの選択とカスタマイズの詳細については、「」を参照してくださいステップ 3: テンプレートを選択してカスタマイズする AWS CloudFormation。

AWS Ground Stationでクロスリージョンのデータ配信を使用するには、次のトピックを使用します。

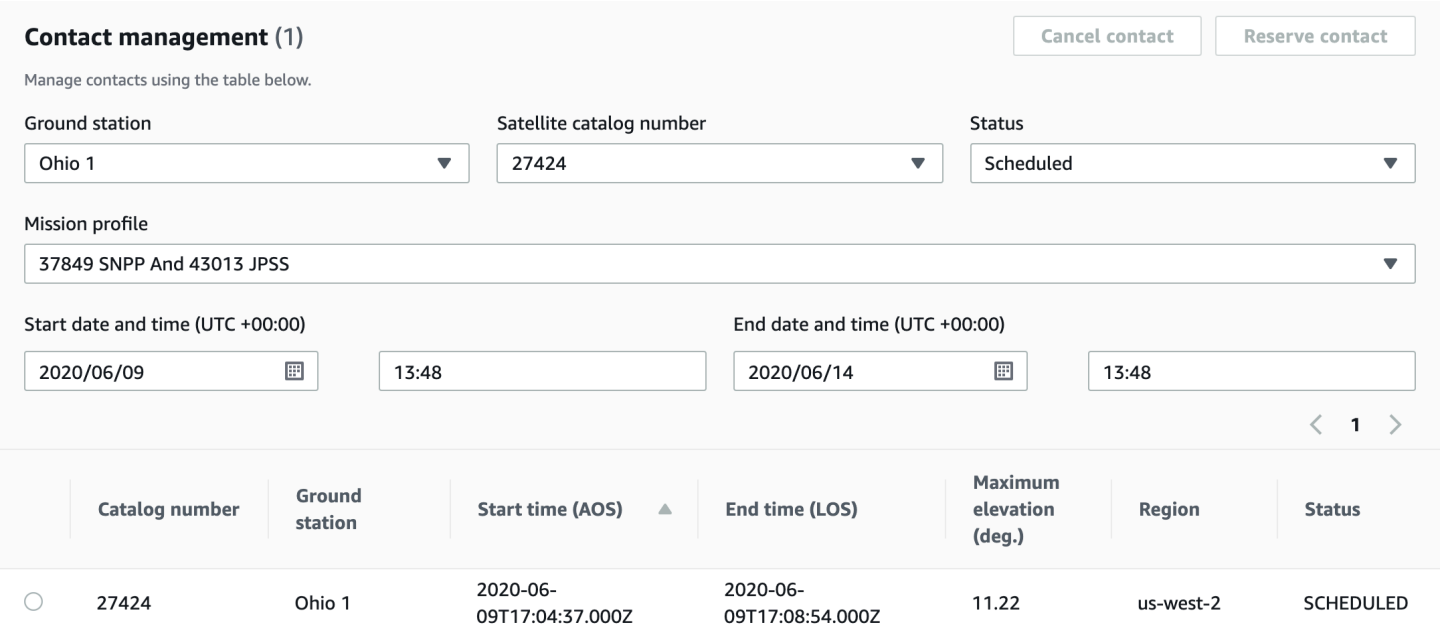
トピック

- コンソールでクロスリージョンのデータ配信を使用するには
- AWS CLI でクロスリージョンのデータ配信を使用するには

コンソールでクロスリージョンのデータ配信を使用するには

AWS Ground Station コンソールで連絡先を予約するときは、連絡先データを目的のリージョンに配信するように設定されたミッションプロファイルを選択します。すべてのパラメータが正しいことを確認し、[Reserve contact (コンタクトの予約)] を選択します。コンソールに目的のミッションプロファイルが表示されない場合は、コンソールを表示しているリージョンでミッションプロファイルを作成したことを確認します。

コンタクトを予約した後、スケジュールされたコンタクトを表示し、地上ステーションアンテナの位置と宛先リージョンを表示することにより、クロスリージョンのデータ配信がスケジュールされていることを確認できます。次の図は、クロスリージョンのデータ配信がスケジュールされているコンタクトを示しています。このコンタクトは、オハイオの地上ステーションのアンテナを使用し、オレゴンにデータを配信するように設定されています。



AWS CLI でクロスリージョンのデータ配信を使用するには

で連絡先を予約するときは AWS CLI、連絡先データを目的のリージョンに配信するように設定されたミッションプロファイルを選択します。`--mission-profile-arn <value>` で目的のミッションプロファイルの ARN を指定します。すべてのパラメータが正しいことを確認し、コマンドを実行します。コンタクトを表示およびリスト化するときに、目的のミッションプロファイル ARN が表示されない場合は、 AWS CLIを実行中のリージョンでミッションプロファイルを作成したことを確認します。

コンタクトを予約した後、スケジュールされたコンタクトを表示し、地上ステーションアンテナの位置と宛先リージョンを表示することにより、クロスリージョンのデータ配信がスケジュールされていることを確認できます。次の出力は、クロスリージョンのデータ配信がスケジュールされているコンタクトを示しています。このコンタクトは、オハイオの地上ステーションのアンテナを使用し、オレゴンにデータを配信するように設定されています。

```
{
  "contactList": [
    {
      "contactId": "11111111-2222-3333-4444-555555555555",
      "contactStatus": "SCHEDULED",
      "endTime": "2020-05-05T03:16:35-06:00",
      "groundStation": "Ohio 1",
      "maximumElevation": {
        "unit": "DEGREE_ANGLE",
```

```
        "value": 26.74
      },
      "missionProfileArn": "arn:aws:groundstation:us-west-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555",
      "postPassEndTime": "2020-05-05T03:17:35-06:00",
      "prePassStartTime": "2020-05-05T03:04:08-06:00",
      "region": "us-west-2",
      "satelliteArn":
 "arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555",
      "startTime": "2020-05-05T03:06:08-06:00"
    }
  ]
}
```

モニタリング AWS Ground Station

モニタリングは、 AWS Ground Stationの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。AWS は、異常を監視して報告し AWS Ground Station、必要に応じて自動アクションを実行するための以下の監視ツールを提供しています。

- Amazon CloudWatch Events は、 AWS リソースの変化を説明するシステムイベントのストリームをほぼリアルタイムで配信します。 CloudWatch Events を使用すると、特定のイベントを監視し、 AWS そのイベントが発生したときに他のサービスで自動アクションをトリガーするルールを記述できるため、イベント主導型のコンピューティングを自動化できます。Amazon イベントの詳細については、Amazon CloudWatch CloudWatch イベントユーザーガイドを参照してください。
- AWS EventBridge Events は、 AWS リソースの変化を説明するシステムイベントのストリームをほぼリアルタイムで配信します。 EventBridge Events を使用すると、特定のイベントを監視し、AWS そのイベントが発生したときに他のサービスで自動アクションをトリガーするルールを記述できるため、イベント主導型のコンピューティングを自動化できます。 EventBridge イベントの詳細については、Amazon EventBridge Events ユーザーガイドを参照してください。
- AWS CloudTrailアカウントによって、 AWS またはアカウントに代わって行われた API 呼び出しと関連イベントをキャプチャし、指定した Amazon S3 バケットにログファイルを配信します。AWSを呼び出したユーザーとアカウント、呼び出し元の IP アドレス、および呼び出し日時を特定できます。詳細については AWS CloudTrail、『AWS CloudTrail ユーザーガイド』を参照してください。
- Amazon CloudWatch Metricsは、 AWS Ground Station使用時にスケジュールされた連絡先のメトリックスをキャプチャします。 CloudWatch メトリックスでは、チャネル、偏波、衛星 ID に基づいてデータを分析し、連絡先の信号強度とエラーを特定できます。詳細については、「Amazon CloudWatch メトリクスの使用」を参照してください。
- AWS User Notifications を使用して、 AWS Ground Station イベントに関する通知を受け取る配信チャネルを設定できます。指定したルールにイベントが一致すると、通知を受け取ります。イベントの通知は、E メール、AWS Chatbot チャット通知、AWS Console Mobile Application プッシュ通知など、複数のチャネルを通じて受け取ることができます。また、コンソール通知センターの通知を確認することもできます。 User Notifications は集約をサポートしているため、特定のイベント中に受け取る通知の数を減らすことができます。

AWS Ground Station をモニタリングするには、次のトピックを参照してください。

トピック

- AWS Ground Station イベントによる自動化
- AWS Ground Station API 呼び出しの口ギング: AWS CloudTrail
- Amazon ンのメトリクス CloudWatch

AWS Ground Station イベントによる自動化

> **Note**
>
> このドキュメントでは、全体を通して「イベント」という用語を使用しています。CloudWatch EventBridge イベントとは、基盤となる同じサービスと API です。いずれかのサービスを使用することで、受信イベントを一致させ、処理のためにターゲットにルーティングするルールを作成できます。

イベントを使用すると、 AWS サービスを自動化し、アプリケーションの可用性に関する問題やリソースの変更などのシステムイベントに自動的に対応できます。 AWS サービスからのイベントはほぼリアルタイムで配信されます。簡単なルールを記述して、注目するイベントと、イベントがルールに一致した場合に自動的に実行するアクションを指定できます。自動的にトリガーできるオペレーションには、以下が含まれます。

- 関数を呼び出す。 AWS Lambda
- Amazon EC2 Run Command の呼び出し
- Amazon Kinesis Data Streams へのイベントの中継
- AWS Step Functions ステートマシンのアクティベーション
- Amazon SNS トピックまたはキューへの通知 AWS SMS

でのイベントの使用例には次のようなものがあります。 AWS Ground Station

- イベント状態に基づいて Amazon EC2 インスタンスの開始と停止を自動化する Lambda 関数を呼び出す。
- コンタクトの状態が変化するたびに Amazon SNS トピックを発行する。これらのトピックは、コンタクトの最初または最後に E メール通知を送信するように設定できます。

詳細については、Amazon CloudWatch イベントユーザーガイドまたは Amazon EventBridge イベントユーザーガイドを参照してください。

イベントの例

> **Note**
>
> AWS Ground Station によって生成されるすべてのイベントには、「ソース」の値として「aws.groundstation」が含まれます。

Ground Station のコンタクト状態の変化

今後コンタクトの状態が変わったときに特定のアクションを実行する場合は、このアクションを自動化する ルールを設定できます。これは、コンタクトの状態変更に関する通知を受信する場合に役立ちます。これらのイベントを受信するタイミングを変更したい場合は、contactPrePassDurationSecondsミッションプロファイルのとを変更できますcontactPostPassDurationSeconds。イベントは、コンタクトのスケジュール元のリージョンに送信されます。

以下に例を示します。

```
{
    "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"
    },
    "account": "123456789012"
```

```
}
```

contactStatus に指定できる値は、the section called “Ground Station コンタクトのステータス” で定義されています。

Ground Station データフローエンドポイントグループの状態変更

データフローエンドポイントグループをデータ受信に使用しているときにアクションを実行する場合は、このアクションを自動化する ルールを設定できます。これにより、データフローエンドポイントグループステータスの状態変更に応じて、さまざまなアクションを実行できます。これらのイベントを受信するタイミングを変更したい場合は、contactPrePassDurationSecondsとが異なるデータフローエンドポイントグループを使用してください。contactPostPassDurationSecondsこのイベントは、データフローエンドポイントグループのリージョンに送信されます。

以下に例を示します。

```
{
    "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"
    },
    "account": "123456789012"
```

```
}
```

dataflowEndpointGroupState の状態として、PREPASS、PASS、POSTPASS、および COMPLETED が考えられます。

Ground Station エフェメリス状態の変化

エフェメリスの状態が変わったときにアクションを実行する場合は、このアクションを自動化するルールを設定できます。これにより、エフェメリスの状態変化に応じてさまざまなアクションを実行できます。例えば、エフェメリスの検証が完了し、ENABLED になっているときにアクションを実行できます。このイベントの通知は、エフェメリスがアップロードされたリージョンに送信されます。

以下に例を示します。

```
    {
    "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"
    }
    }
```

ephemerisStatus の状態として、ENABLED、VALIDATING、INVALID、ERROR、DISABLED、および EXPIRED が考えられます。

AWS Ground Station API 呼び出しの口ギング: AWS CloudTrail

AWS Ground Station 内のユーザ AWS CloudTrail、ロール、 AWS またはサービスが行ったアクションの記録を提供するサービスと統合されている AWS Ground Station。 CloudTrail すべての API

AWS Ground Station 呼び出しをイベントとしてキャプチャします。キャプチャされた呼び出しには、 AWS Ground Station コンソールからの呼び出しと AWS Ground Station API オペレーションへのコード呼び出しが含まれます。証跡を作成すると、 CloudTrail のイベントを含むイベントを Amazon S3 バケットに継続的に配信できるようになります AWS Ground Station。証跡を設定しなくても、 CloudTrail コンソールの [イベント履歴] で最新のイベントを確認できます。によって収集された情報を使用して CloudTrail、要求の送信元 IP アドレス AWS Ground Station、要求の実行者、実行日時、その他の詳細情報を確認できます。

詳細については CloudTrail、『AWS CloudTrail ユーザーガイド』を参照してください。

AWS Ground Station の情報 CloudTrail

CloudTrail アカウントを作成すると、 AWS アカウントで有効になります。でアクティビティが発生すると AWS Ground Station、 CloudTrail AWS そのアクティビティはイベント履歴の他のサービスイベントとともにイベントに記録されます。 AWS アカウント内の最近のイベントを表示、検索、ダウンロードできます。詳細については、「 CloudTrail イベント履歴によるイベントの表示」を参照してください。

AWS アカウント内のイベント（のイベントを含む）を継続的に記録するには AWS Ground Station、トレイルを作成してください。トレイルを使用すると CloudTrail 、Amazon S3 バケットにログファイルを配信できます。デフォルトでは、コンソールで作成した証跡がすべての AWS リージョンに適用されます。トレイルは、 AWS パーティション内のすべてのリージョンからのイベントを記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、 AWS CloudTrail ログに収集されたイベントデータをさらに分析して処理するように他のサービスを設定できます。詳細については、次を参照してください:

- 証跡の作成のための概要
- CloudTrail サポート対象のサービスとインテグレーション
- の Amazon SNS 通知の設定 CloudTrail
- CloudTrail 複数のリージョンからのログファイルの受信、 CloudTrail 複数のアカウントからのログファイルの受信

AWS Ground Station すべてのアクションは AWS Ground Station API CloudTrail リファレンスによって記録され、文書化されています。たとえば、、を呼び出したりReserveContact、CancelContactListConfigsアクションを実行したりすると、 CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます:

- リクエストが root ユーザー認証情報または AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- AWS リクエストが別のサービスによって行われたかどうか。

詳細については、「CloudTrail userIdentity エレメント」を参照してください。

AWS Ground Station ログファイルエントリについて

トレイルは、指定した Amazon S3 バケットにイベントをログファイルとして配信できるようにする設定です。 CloudTrail ログファイルには 1 つ以上のログエントリが含まれます。イベントはあらゆるソースからの単一のリクエストを表し、リクエストされたアクションに関する情報、アクションの日時、リクエストパラメータなどが含まれます。 CloudTrail ログファイルはパブリック API 呼び出しの順序付けられたスタックトレースではないため、特定の順序で表示されることはありません。

次の例は、 CloudTrail `ReserveContact`アクションを示すログエントリを示しています。

例: ReserveContact

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINICIPLE_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_PRINICIPLE_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": "Coral/Jakarta",
    "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"
}
```


Amazon ンのメトリクス CloudWatch

問い合わせ中、 AWS Ground Station CloudWatch データを自動的にキャプチャして分析用に送信します。データは、Amazon CloudWatch コンソールのグラフまたはソースコードとして表示できます。 CloudWatch アクセスとメトリックスについて詳しくは、「Amazon CloudWatch メトリクスの使用」を参照してください。

AWS Ground Station メトリックスとディメンション

利用可能なメトリクス

以下の指標はから入手できます AWS Ground Station。

メトリクス	説明
AzimuthAngle	アンテナの方位角。真北は 0 度、東は 90 度です。 単位: 度
BitErrorRate	伝送したビットのうち、エラーが発生したビットの割合。ビットエラーは、ノイズ、ゆがみ、または干渉によって発生します。 単位: 単位時間あたりのビットエラー数
BlockErrorRate	受信したブロックのうち、エラーが発生したブロックの割合。ブロックエラーは干渉によって発生します。 単位: エラーが発生したブロック数/ブロック総数
CarrierFrequencyRecovery_Cn0	単位帯域幅あたりのキャリア対ノイズ密度の比率。 単位: デシベルヘルツ (dB-Hz)
CarrierFrequencyRecovery_Locked	復調器のキャリア周波数回復ループがロックされている場合は 1 に設定され、ロックが解除されている場合は 0 に設定されます。 単位: 単位なし
CarrierFrequencyRecovery_OffsetFrequency_Hz	推定された信号中心周波数と理想的な中心周波数の間のオフセット。この原因は、宇宙機とアンテナシステム間のドップラーシフトと局部発振器のオフセットです。 単位: ヘルツ (Hz)
ElevationAngle	アンテナの仰角。水平線は 0 度、天頂は 90 度です。 単位: 度

メトリクス	説明
Es/N0	シンボルあたりのエネルギーとノイズパワースペクトル密度の比率。 単位: デシベル (dB)
ReceivedPower	復調器/デコーダで測定された信号強度。 単位: ミリワットを基準値とするデシベル (dBm)
SymbolTimingRecovery_ErrorVectorMagnitude	受信したシンボルと理想的なコンスタレーション点の間の誤差ベクトルの大きさ。 単位: パーセント
SymbolTimingRecovery_Locked	復調器シンボルのタイミング回復ループがロックされている場合は 1 に設定され、ロックが解除されている場合は 0 に設定されます。 単位: 単位なし
SymbolTimingRecovery_OffsetSymbolRate	推定シンボルレートと理想的な信号シンボルレートの間のオフセット。この原因は、宇宙機とアンテナシステム間のドップラーシフトと局部発振器のオフセットです。 単位: シンボル/秒

AWS Ground Stationどのディメンションが使われているのか？

AWS Ground Station 次のディメンションを使用してデータをフィルタリングできます。

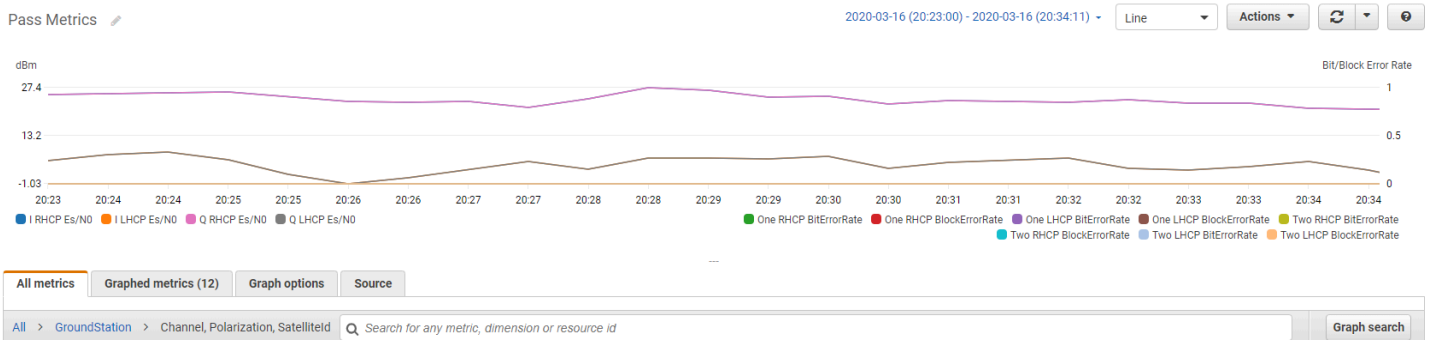
ディメンション	説明
Channel	各コンタクトのチャネルには、1、2、I (同相)、Q (直交) があります。

ディメンション	説明
Polarization	各コンタクトの偏波には、LHCP (左円偏波) または RHCP (右円偏波) があります。
SatelliteId	人工衛星 ID には、コンタクトの人工衛星の ARN が含まれます。

メトリクスの表示

グラフ化されたメトリクスを表示する場合、集計の時間帯によってメトリクスの表示方法が変わることに注意する必要があります。データの受信後 3 時間の間は、コンタクトの各メトリクスが 1 秒あたりのデータとして表示されます。3 時間が経過すると、データは 1 分あたりのデータとして CloudWatch Metrics 別に集計されます。1 秒あたりのデータ測定値に関するメトリクスを表示する必要がある場合は、データを受信してから 3 時間以内にデータを表示するか、メトリクスの外部に保存することをおすすめします。 CloudWatch

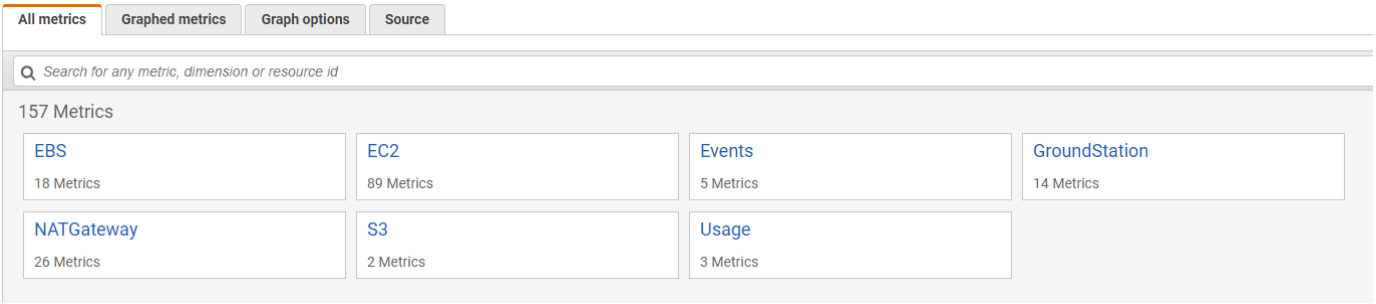
さらに、最初の 60 秒以内にキャプチャされたデータには、意味のあるメトリクスを生成するための十分な情報が含まれていないため、データが表示されない可能性があります。意味のあるメトリクスを表示するには、60 秒が経過した後でデータを表示することをお勧めします。



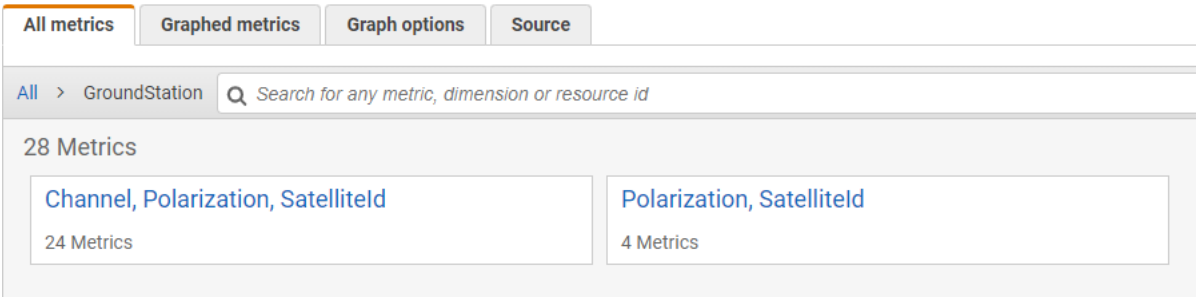
AWS Ground Station での指標のグラフ化について詳しくは CloudWatch、「指標のグラフ化」を参照してください。

コンソールを使用してメトリクスを表示するには

1. CloudWatch コンソールを開きます。
2. ナビゲーションペインで メトリクスを選択します。
3. GroundStation 名前空間を選択します。



4. 必要なメトリクスディメンション (チャネル、偏光など) を選択します。 SatelliteId



5. [All metrics] タブには、名前空間内のそのディメンションのメトリクスがすべて表示されます。以下の操作を行うことができます。

   a. テーブルを並べ替えるには、列見出しを使用します。

   b. メトリクスをグラフ化するには、そのメトリクスに対応するチェックボックスをオンにします。すべてのメトリクスを選択するには、テーブルの見出し行にあるチェックボックスを選択します。

   c. リソースでフィルタするには、リソース ID を選択し、[Add to search] を選択します。

   d. メトリクスでフィルタするには、メトリクスの名前を選択し、[Add to search] を選択します。

を使用してメトリクスを表示するには AWS CLI

1. AWS CLI がインストールされていることを確認します。インストールの詳細については AWS CLI、「AWS CLI のインストール」を参照してください。
2. CloudWatch エージェント設定 JSON ファイルを作成します。 CloudWatch エージェント設定ファイルの作成方法については、「 CloudWatch エージェント設定ファイルの作成」を参照してください。

3. CloudWatch を実行して利用可能なメトリクスを一覧表示しますaws cloudwatch list-metrics。
4. ステップ 2 で作成した JSON ファイルを変更して、メトリクスの SatellitID と一致させます。

> **Note**
>
> Periodこのフィールドを 60 未満の値に減らさないでください。 AWS Ground Station 60 秒ごとにメトリクスを公開し、値を減らしてもメトリクスは返されません。

5. CloudWatch パスの期間とエージェント設定 JSON aws cloudwatch get-metric-data ファイルを使用して実行します。以下に例を示します。

```
aws cloudwatch get-metrics-data -start-time 2020-02-26T19:12:00Z -end-time
 2020-02-26T19:24:00Z -metric-data-queries file://metricdata.json
```

メトリクスには、コンタクトのタイムスタンプが表示されます。 AWS Ground Station メトリクスの出力例を以下に示します。

```
{
      “MetricDataResults”: [
          {
              “Id”: “myQuery”,
              “Label”: “Es/N0”,
              “Timestamps”: [
                  “2020-02-18T19:44:00Z”,
                  “2020-02-18T19:43:00Z”,
                  “2020-02-18T19:42:00Z”,
                  “2020-02-18T19:41:00Z”,
                  “2020-02-18T19:40:00Z”,
                  “2020-02-18T19:39:00Z”,
                  “2020-02-18T19:38:00Z”,
                  “2020-02-18T19:37:00Z”,
              ],
              “Values”: [
                  24.58344556958329,
                  24.251638725562216,
                  22.919391450230158,
                  22.83838908204037,
                  23.303086848486842,
```

```
                    22.845261784583364,
                    21.34531397048953,
                    19.171561698261222
                ],
                “StatusCode”: “Complete”
            }
        ]
        “Messages”: []
}
```

トラブルシューティング

次のドキュメントは、 AWS Ground Station 連絡が正常に完了しない可能性がある問題のトラブルシューティングに役立ちます。

トピック

Amazon EC2 にデータを配信するコンタクトのトラブルシューティング

AWS Ground Station 問い合わせを正常に完了できない場合は、Amazon EC2 インスタンスが実行中であること、Data Defender が実行中であること、および Data Defender ストリームが適切に設定されていることを確認する必要があります。

前提条件

次の手順では、Amazon EC2 インスタンスがすでにセットアップされていることを前提としています。で Amazon EC2 インスタンスをセットアップするには AWS Ground Station、「はじめに」を参照してください。

ステップ 1: EC2 インスタンスが実行されているか確認する

1. トラブルシューティングする連絡先に使用された Amazon EC2 インスタンスを見つけます。以下のステップを使用します。

   a. CloudFormationダッシュボードで、Amazon EC2 インスタンスを含むスタックを選択します。

   b. [リソース] タブをクリックし、Amazon EC2 インスタンスを [論理 ID] 列でロードバランサーの ID をクリックします。[状況] 列でインスタンスが作成されていることを確認します。

   c. [物理 ID] 列で、Amazon EC2 インスタンスのリンクを選択します。Amazon EC2 マネジメントコンソールが表示されます。

2. Amazon EC2 マネジメントコンソールで、Amazon EC2 の [インスタンスの状態] が [実行中] になっていることを確認します。
3. インスタンスが実行中の場合は、次のステップに進みます。インスタンスが実行されていない場合は、次の手順を使用してインスタンスを起動します。

    - Amazon EC2 インスタンスを選択した状態で、[アクション] > [インスタンスの状態] > [開始] の順に選択します。

ステップ 2: 使用するデータフローアプリケーションのタイプを判別する

データ配信に AWS Ground Station Agent を使用している場合は、「エージェントのトラブルシューティング AWS Ground Station」セクションにリダイレクトしてください。

それ以外の場合、Data Defender (DDX) アプリケーションを使用していれば、「the section called “ステップ 3: Data Defender が実行されているか確認する”」に進んでください。

ステップ 3: Data Defender が実行されているか確認する

Data Defender のステータスを確認するには、Amazon EC2 でインスタンスに接続する必要があります。インスタンスへの接続の詳細については、「Linux インスタンスへの接続」を参照してください。

次の手順では、SSH クライアントでコマンドを使用したトラブルシューティングの手順を示します。

1. ターミナルまたはコマンドプロンプトを開き、SSH を使用して Amazon EC2 インスタンスに接続します。Data Defender Web UI を表示するために、リモートホストのポート 80 を転送します。以下のコマンドは、SSH を使用して、ポート転送が有効になっている踏み台を介して Amazon EC2 インスタンスに接続する方法を示しています。

    > **Note**
    >
    > <SSH KEY>、<BASTION HOST>、および <HOST> は、特定の ssh キー、踏み台ホスト名、および Amazon EC2 インスタンスホスト名に置き換える必要があります。

    Windows の場合

```
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>
```

Mac の場合

```
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. 出力で ddx という名前の実行中のプロセスをグレッピング (チェック) して、Data Defender (DDX とも呼ばれる) が実行されていることを確認します。実行中のプロセスをグレッピング (チェック) するためのコマンドと成功した出力例を以下に示します。

```
[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
```

Data Defender が実行中の場合は、「the section called “ステップ 4: Data Defender Stream が設定されていることを確認する”」に進みます。そうでない場合は、次のステップに進みます。

3. 次のコマンドを使用して Data Defender を起動します。

```
sudo service rtlogic-ddx start
```

このコマンドの使用後に Data Defender が実行中になった場合は、「the section called “ステップ 4: Data Defender Stream が設定されていることを確認する”」に進みます。そうでない場合は、次のステップに進みます。

4. Data Defender のインストールおよび設定中にエラーが発生したかどうかを確認するには、以下のコマンドを使用して以下のファイルを検査します。

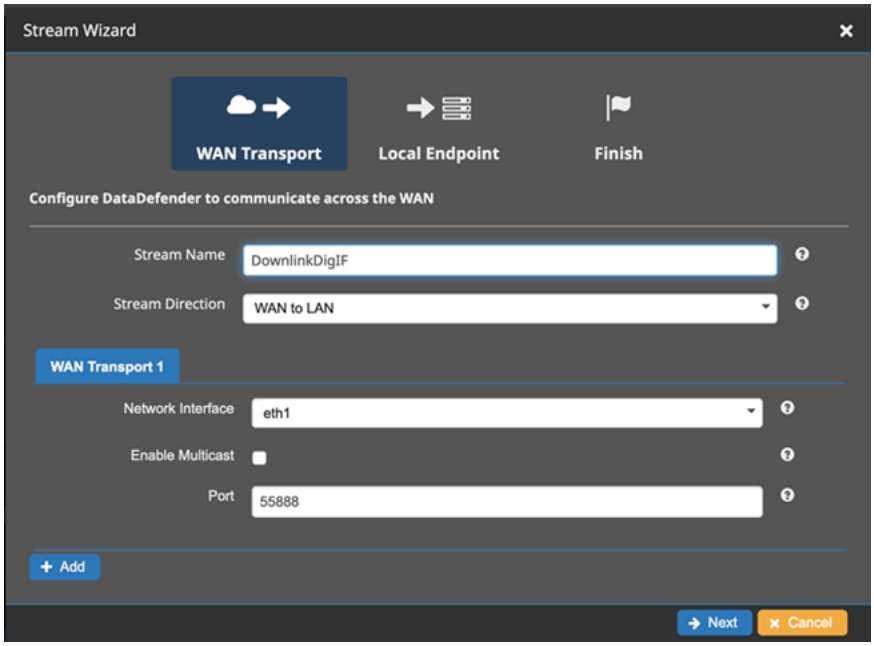
```
cat /var/log/user-data.log
cat /opt/aws/groundstation/.startup.out
```

> **Note**
>
> これらのファイルを検査したときに発見される一般的な問題は、Amazon EC2 インスタンスが実行されている Amazon VPC に Amazon S3 へのアクセス許可がないためにインストールファイルをダウンロードできないことです。これが問題であることがログで判明した場合は、EC2 インスタンスの Amazon VPC とセキュリティグループの設定をチェックして、Amazon S3 へのアクセスがブロックされていないことを確認します。

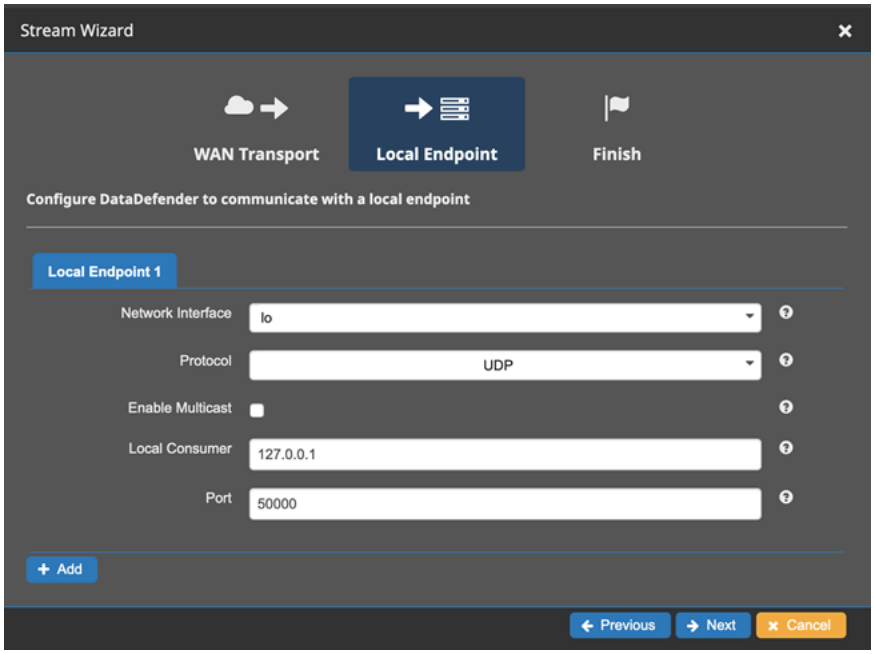
Amazon VPC 設定を確認した後で Data Defender が実行中になった場合は、「the section called “ステップ 4: Data Defender Stream が設定されていることを確認する”」に進みます。問題が解決しない場合は、AWS サポートに連絡し、問題の説明を添えてログファイルを送信してください。

ステップ 4: Data Defender Stream が設定されていることを確認する

1. ウェブブラウザで、アドレスバーに localhost:8080 というアドレスを入力して、DDX Web User Interface にアクセスします。次に、<Enter> キーを押します。
2. DataDefenderダッシュボードで [詳細に移動] を選択します。
3. ストリームのリストからストリームを選択し、[Edit Stream (ストリームを編集)] を選択します。
4. [Stream Wizard (ストリームウィザード)] ダイアログボックスで、次の操作を行います。
   a. [WAN Transport] (WAN トランスポート) ペインで、[Stream Direction] (ストリーム方向) が [WAN to LAN] (WAN から LAN) に設定されていることを確認します。
   b. [Port (ポート)] ボックスで、データフローエンドポイントグループ用に選択した WAN ポートが存在することを確認します。デフォルトでは、このポートは 55888 です。[次へ] を選択します。



c. [Local Endpoint (ローカルエンドポイント)] ペインで、[Port (ポート)] ボックスに有効なポートがあることを確認します。デフォルトでは、このポートは 50000 です。Data Defender AWS Ground Station がサービスからデータを受信した後に、このポートでデータを受信します。[次へ] を選択します。



d. 値を変更した場合は、残りのメニューで [Finish (完了)] を選択します。それ以外の場合は、[Stream Wizard (ストリームウィザード)] メニューからキャンセルできます。

これで、Amazon EC2 インスタンスと Data Defender の両方が実行され、からデータを受信するように正しく設定されていることを確認できました。 AWS Ground Station問題が解決しない場合は、AWS サポートにお問い合わせください。

Ground Station コンタクトのステータス

AWS Ground Station 連絡先のステータスから、特定の時間にその連絡先に何が起こっているかを知ることができます。

コンタクトのステータス

コンタクトに設定できるステータスのリストは次のとおりです。

- 利用可能 - コンタクトが予約可能です。
- SCHEDULING - コンタクトはスケジュール設定中です。
- SCHEDULED - コンタクトが正常にスケジュール設定されました。
- FAILED_TO_SCHEDULE - コンタクトがスケジュール設定に失敗しました。
- PREPASS - コンタクトがまもなく開始され、リソースを準備中です。
- PASS - コンタクトが現在実行中で、衛星と通信中です。
- POSTPASS - 通信が完了し、使用中のリソースをクリーンアップ中です。
- COMPLETED - コンタクトが正常に完了しました。
- FAILED - お客様リソース設定の問題のためにコンタクトに失敗しました。
- AWS_FAILED-サービスの問題が原因でコンタクトが失敗しました。 AWS Ground Station
- cancelling - コンタクトがキャンセルのプロセス中です。
- AWS_CANCELED-コンタクトはサービスによってキャンセルされました。 AWS Ground Station これが生じる可能性がある例としては、アンテナやサイトのメンテナンスがあります。
- CANCELLED - コンタクトがお客様によってキャンセルされました。

トラブルシューティングガイド

- the section called “FAILED になったコンタクトのトラブルシューティング”
- the section called “FAILED_TO_SCHEDULE 連絡先のトラブルシューティング”

FAILED になったコンタクトのトラブルシューティング

AWS Ground Station 顧客リソース設定に問題が検出されると、連絡先の端末連絡先ステータスは FAILED になります。コンタクトが FAILED になる原因となる一般的な使用例と、トラブルシューティングに役立つ手順を以下に示します。

> **Note**
>
> このガイドは、特にコンタクトの FAILED ステータスを対象としており、AWS_FAILED、AWS_CANCELED、FAILED_TO_SCHEDULE などの他の失敗ステータスを対象としたものではありません。コンタクトのステータスの詳細については、「the section called “Ground Station コンタクトのステータス”」を参照してください。

Data Defender (DDX) の FAILED のユースケース

DDX ベースのデータフローでコンタクトステータスが FAILED になることがある一般的なユースケースを、以下に示します。

- カスタマー DDX が接続されない-1 つ以上のデータフローについて、 AWS Ground Station アンテナとカスタマーデータフローエンドポイントグループ間の DDX 接続が確立されませんでした。
- カスタマー DDX 接続が遅い-1 つ以上のデータフローの AWS Ground Station Antenna とカスタマーデータフローエンドポイントグループ間の DDX 接続が、コンタクト開始時刻以降に確立されました。

DDX データフローに失敗が発生した場合は、次の点を確認することをお勧めします。

- コンタクト開始時刻より前に、受信側の Amazon EC2 インスタンスが正常に起動したことを確認します。
- コンタクト中に DDX が起動して実行されていたことを確認します。

より具体的なトラブルシューティング手順については、「the section called “Amazon EC2 にデータを配信するコンタクトのトラブルシューティング”」のセクションを参照してください。

AWS Ground Station エージェントが失敗したユースケース

Agent ベースのデータフローでコンタクトステータスが FAILED になることがある一般的なユースケースを、以下に示します。

- カスタマーエージェントの未報告ステータス-カスタマーデータフローエンドポイントグループで、ステータスを報告できなかったデータフローの 1 つまたは複数のデータフローのデータ配信を調整する担当エージェント。 AWS Ground Stationこのステータスの更新は、コンタクト終了時刻の数秒以内に行われます。
- カスタマーエージェントの起動が低速である – 1 つ以上のデータフローのカスタマーデータフローエンドポイントグループで、データ配信のオーケストレーションを担当するエージェントの開始が、問い合わせの開始時刻を過ぎて遅くなった。

AWS Ground Station Agent データフローで障害が発生した場合は、以下を確認することをお勧めします。

- コンタクト開始時刻より前に、受信側の Amazon EC2 インスタンスが正常に起動したことを確認します。
- コンタクトの開始時とコンタクト中に、Agent アプリケーションが起動して実行中であったことを確認します。
- Agent アプリケーションと Amazon EC2 インスタンスが、コンタクト終了から 15 秒以内にシャットダウンされていないことを確認します。これにより、Agent は AWS Ground Stationにステータスを報告するのに十分な時間を確保できます。

より具体的なトラブルシューティング手順については、「the section called “Amazon EC2 にデータを配信するコンタクトのトラブルシューティング”」のセクションを参照してください。

FAILED_TO_SCHEDULE 連絡先のトラブルシューティング

AWS Ground Station 顧客のリソース構成または内部システム内で問題が検出されると、連絡先は FAILED_TO_SCHEDULE になります。FAILED_TO_SCHEDULE 状態で終わる連絡先は、オプションで追加のコンテキストを提供します。`errorMessage`連絡先の説明については、を参照してください。the section called “連絡先を記述してください AWS CLI”

FAILED_TO_SCHEDULE コンタクトの原因となる一般的な使用例と、トラブルシューティングに役立つ手順を以下に示します。

> **Note**
>
> このガイドは FAILED_TO_SCHEDULE コンタクトステータスを対象としており、AWS_FAILED、AWS_CANCELED、FAILED などの他の障害ステータスを対象としたものではありません。コンタクトのステータスの詳細については、「the section called “Ground Station コンタクトのステータス”」を参照してください。

アンテナダウンリンクデモデコード設定で指定されているConfig はサポートされていません

この連絡先のスケジュールに使用されたミッションプロファイルには、antenna-downlink-demod-decode 無効な構成が含まれていました。

AntennaDownlinkDemodDecode 以前から存在していたコンフィグ

- antenna-downlink-demod-decode 構成が最近変更された場合は、スケジュールを立てる前に、以前に動作していたバージョンにロールバックしてください。
- これが既存の構成を意図的に変更した場合や、スケジュールが正常に行われていない既存の構成の場合は、次の手順に従って新しい構成をオンボーディングしてください。AntennaDownlinkDemodDecode

新しく作成された設定 AntennaDownlinkDemodDecode

新しいコンフィグをオンボーディングするには、 AWS Ground Station 直接お問い合わせください。FAILED_TO_SCHEDULE **`contactId`** 状態で終了したケースを含め、AWS Support でケースを作成してください

一般的なトラブルシューティングステップ

前述のトラブルシューティング手順で問題が解決しなかった場合:

- 連絡先のスケジュールを再試行するか、同じミッションプロファイルを使用して別の連絡先をスケジュールしてください。the section called “連絡先を予約してください AWS CLI” を参照してください。
- このミッションプロファイルの FAILED_TO_SCHEDULE ステータスが引き続き表示される場合は、AWS Support にお問い合わせください

AWS Ground Station のセキュリティ

AWS では、クラウドセキュリティが最優先事項です。AWS のお客様は、セキュリティを最も重視する組織の要件を満たすよう構築されたデータセンターとネットワークアーキテクチャを利用できます。AWS は、セキュリティの目標達成に役立つセキュリティ固有のツールと機能を提供しています。これらのツールと機能には、ネットワークセキュリティ、設定管理、アクセスコントロール、およびデータセキュリティが含まれます。

AWS Ground Station を使用する場合は、業界のベストプラクティスに従い、エンドツーエンドの暗号化を実装することをお勧めします。AWS では、暗号化とデータ保護を統合するための API を提供しています。AWS セキュリティの詳細については、「AWS セキュリティの紹介」ホワイトペーパーを参照してください。

以下のトピックでは、 のリソースをセキュリティで保護する方法について説明します。

トピック

- AWS Ground Station 向けの Identity and Access Management
- Ground Station のサービスにリンクされたロールの使用
- AWS の AWS Ground Station マネージドポリシー

AWS Ground Station 向けの Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御するために役立つ AWS のサービスです。IAM 管理者は、誰をわ認証 (サインイン) し、誰に AWS Ground Station リソースの使用を許可する (権限を持たせる) かを制御します。IAM は、無料で使用できる AWS のサービスです。

トピック

- 対象者
- アイデンティティを使用した認証
- ポリシーを使用したアクセス権の管理
- AWS Ground Station と IAM の連携方法
- AWS Ground Station のアイデンティティベースのポリシーの例
- AWS Ground Station ID とアクセスのトラブルシューティング

対象者

AWS Identity and Access Management (IAM) の用途は、AWS Ground Station で行う作業によって異なります。

サービスユーザー - AWS Ground Station サービスを使用してジョブを実行する場合は、必要な権限と認証情報を管理者が用意します。作業を実行するためにさらに多くの AWS Ground Station 機能を使用するとき、追加の権限が必要になる場合があります。アクセスの管理方法を理解すると、管理者から適切な権限をリクエストするのに役に立ちます。AWS Ground Station 機能にアクセスできない場合は、「AWS Ground Station ID とアクセスのトラブルシューティング」を参照してください。

サービス管理者 - 社内の AWS Ground Station リソースを担当している場合は、通常、AWS Ground Station への完全なアクセスがあります。サービスのユーザーがどの AWS Ground Station 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を確認して、IAM の基本概念を理解してください。お客様の会社で AWS Ground Station で IAM を利用する方法の詳細については、「AWS Ground Station と IAM の連携方法」を参照してください。

IAM 管理者 - 管理者は、AWS Ground Station へのアクセスを管理するポリシーの書き込み方法の詳細について確認する場合があります。IAM で使用できる AWS Ground Station アイデンティティベースのポリシーの例を表示するには、「AWS Ground Station のアイデンティティベースのポリシーの例」を参照してください。

アイデンティティを使用した認証

認証とは、アイデンティティ認証情報を使用して AWSにサインインする方法です。ユーザーは、AWS アカウントのルートユーザーもしくは IAM ユーザーとして、または IAM ロールを引き受けることによって、認証を受ける (AWS にサインインする) 必要があります。

ID ソースから提供された認証情報を使用して、フェデレーティッドアイデンティティとして AWS にサインインできます。 AWS IAM Identity Centerフェデレーティッドアイデンティティの例としては、(IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報などがあります。フェデレーティッドアイデンティティとしてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーションを使用して AWSにアクセスする場合、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、AWS Management Console または AWS アクセスポータルにサインインできます。AWS へのサインインの詳細については、『AWS サインイン ユーザーガイド』の「AWS アカウント にサインインする方法」を参照してください。

プログラムで AWS にアクセスする場合、AWS は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) を提供し、認証情報でリクエストに暗号で署名します。AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。リクエストに署名する推奨方法の使用については、『IAM ユーザーガイド』の「AWS API リクエストの署名」を参照してください。

使用する認証方法を問わず、追加のセキュリティ情報の提供が求められる場合もあります。例えば、AWS では、アカウントのセキュリティ強化のために多要素認証 (MFA) の使用をお勧めしています。詳細については、「AWS IAM Identity Center ユーザーガイド」の「多要素認証」および「IAM ユーザーガイド」の「AWS での多要素認証 (MFA) の使用」を参照してください。

AWS アカウントのルートユーザー

AWS アカウントを作成する場合は、そのアカウントのすべての AWS のサービスとリソースに対して完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。このアイデンティティは AWS アカウントのルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることによってアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

フェデレーティッド ID

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに対し、ID プロバイダーとのフェデレーションを使用して、一時的な認証情報の使用により、AWS のサービスにアクセスすることを要求します。

フェデレーテッドアイデンティティ は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリのユーザーか、または ID ソースから提供された認証情報を使用して AWS のサービス にアクセスするユーザーです。フェデレーテッドアイデンティティ が AWS アカウント にアクセスすると、ロールが継承され、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、AWS IAM Identity Center を使用することをお勧めします。IAM アイデンティティセンター でユーザーとグループを作成するか、すべての AWS アカウント とアプリケーションで使用するために、独自の ID ソースで一連のユーザーとグループに接続して同期することもできます。IAM アイデンティティセンター の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM アイデンティティセンター?」(IAM アイデンティティセンター とは) を参照してください。

IAM ユーザーとグループ

IAM ユーザーは、1 人のユーザーまたは 1 つのアプリケーションに対して特定の権限を持つ AWS アカウント 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する権限を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、『IAM ユーザーガイド』の「IAM ユーザー (ロールではなく) の作成が適している場合」を参照してください。

IAM ロール

IAM ロールは、特定の権限を持つ、AWS アカウント 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロールを切り替えることによって、AWS Management Console で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、AWS CLI または AWS API オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、『IAM ユーザーガイド』の「IAM ロールの使用」を参照してください。

一時的な認証情報を持った IAM ロールは、以下の状況で役立ちます。

- フェデレーションユーザーユーザーアクセス – フェデレーションアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーティッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー向けロールの作成」を参照してください。IAM アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるもの

を制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。権限セットの詳細については、『AWS IAM Identity Center ユーザーガイド』の「権限セット」を参照してください。

- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の AWS のサービスでは、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、『IAM ユーザーガイド』の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。
- クロスサービスアクセス - 一部の AWS のサービスでは、他の AWS のサービスの機能を使用します。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの権限、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して AWS でアクションを実行するユーザーは、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、AWS のサービスを呼び出すプリンシパルの権限を、AWS のサービスのリクエストと合わせて使用し、ダウンストリームのサービスに対してリクエストを行います。FAS リクエストは、サービスが、完了するために他の AWS のサービス またはリソースとのやりとりを必要とするリクエストを受け取ったときにのみ行われます。この場合、両方のアクションを実行するための権限が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。
  - サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、『IAM ユーザーガイド』の「AWS のサービス に権限を委任するロールの作成」を参照してください。
  - サービスリンクロール - サービスリンクロールは、AWS のサービス にリンクされたサービスロールの一種です。サービスがロールを引き受け、ユーザーに代わってアクションを実行できるようになります。サービスリンクロールは、AWS アカウント に表示され、サービスによって所有されます。IAM 管理者は、サービスリンクロールの権限を表示できますが、編集することはできません。

- Amazon EC2 で実行されているアプリケーション - EC2 インスタンスで実行され、AWS CLI または AWS API 要求を行っているアプリケーションの一時的な認証情報を管理するには、IAM ロールを使用できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスに添付されたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用してアクセス許可を付与する」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、『IAM ユーザーガイド』の「(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?」を参照してください。

ポリシーを使用したアクセス権の管理

AWS でアクセス権を管理するには、ポリシーを作成して AWS アイデンティティまたはリソースにアタッチします。ポリシーは AWS のオブジェクトであり、アイデンティティやリソースに関連付けて、これらの権限を定義します。AWS は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うと、これらのポリシーを評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。大半のポリシーは JSON ドキュメントとして AWSに保存されます。JSON ポリシードキュメントの構造と内容の詳細については、『IAM ユーザーガイド』の「JSON ポリシー概要」を参照してください。

管理者は AWSJSON ポリシーを使用して、だれが何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの権限を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。このポリシーがあるユーザーは、AWS Management Console、AWS CLI、または AWS API からロール情報を取得できます。

アイデンティティベースポリシー

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザー

とロールが実行できるアクション、リソース、および条件を制御します。アイデンティティベースのポリシーを作成する方法については、『IAM ユーザーガイド』の「IAM ポリシーの作成」を参照してください。

アイデンティティベースポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれます。管理ポリシーは、AWS アカウント 内の複数のユーザー、グループ、およびロールにアタッチできるスタンドアロンポリシーです。マネージドポリシーには、AWS マネージドポリシーとカスタマー管理ポリシーがあります。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、『IAM ユーザーガイド』の「マネージドポリシーとインラインポリシーの比較」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーションユーザー、または AWS のサービス を含めることができます。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは IAM の AWSマネージドポリシーは使用できません。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための権限を持つかをコントロールします。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Simple Storage Service (Amazon S3)、AWS WAF、および Amazon VPC は、ACL をサポートするサービスの例です。ACL の詳細については、『Amazon Simple Storage Service デベロッパーガイド』の「アクセスコントロールリスト (ACL) の概要」を参照してください。

その他のポリシータイプ

AWS では、他の一般的ではないポリシータイプをサポートしています。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- 権限の境界 - 権限の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる許可の上限を設定する高度な機能です。エンティティに権限の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとその権限の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、権限の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。権限の境界の詳細については、『IAM ユーザーガイド』の「IAM エンティティの権限の境界」を参照してください。
- サービスコントロールポリシー (SCP) - SCP は、AWS Organizations で組織や組織単位 (OU) の最大権限を指定する JSON ポリシーです。AWS Organizations は、顧客のビジネスが所有する複数の AWS アカウント をグループ化し、一元的に管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP はメンバーアカウントのエンティティに対する権限を制限します (各 AWS アカウントのルートユーザー など)。Organizations と SCP の詳細については、『AWS Organizations ユーザーガイド』の「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限の範囲は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「セッションポリシー」をご参照ください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。複数のポリシータイプが関連するとき、リクエストを許可するかどうかを AWS が決定する方法の詳細については、「IAM ユーザーガイド」の「ポリシーの評価論理」を参照してください。

AWS Ground Station と IAM の連携方法

IAM を使用して AWS Ground Station へのアクセスを管理する前に、AWS Ground Station で利用できる IAM の機能について学びます。

AWS Ground Station で使用できる IAM の機能

IAM の機能	AWS Ground Station サポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	いいえ
ポリシーアクション	あり
ポリシーリソース	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	はい
一時的な認証情報	あり
プリンシパル権限	あり
サービスロール	いいえ
サービスリンクロール	はい

AWS Ground Station およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要を把握するには、IAM ユーザーガイドのIAM と連携する AWS のサービスを参照してください。

AWS Ground Station のアイデンティティベースのポリシー

アイデンティティベースポリシーをサポートする	あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件を制御します。アイデンティティベースのポリシーを作成する方法については、『IAM ユーザーガイド』の「IAM ポリシーの作成」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それがアタッチされているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素について学ぶには、IAM ユーザーガイドのIAM JSON ポリシーの要素のリファレンスを参照してください。

AWS Ground Station のアイデンティティベースのポリシーの例

AWS Ground Station アイデンティティベースのポリシーの例を表示するには、「AWS Ground Station のアイデンティティベースのポリシーの例」を参照してください。

AWS Ground Station 内のリソースベースのポリシー

リソースベースのポリシーのサポート	なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーションユーザー、または AWS のサービス を含めることができます。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM エンティティを、リソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる AWS アカウント にある場合、信頼できるアカウントの IAM 管理者は、リソースにアクセスするための権限をプリンシパルエンティティ (ユーザーまたはロール) に付与する必要もあります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーを追加する必要はありません。詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

AWS Ground Station のポリシーアクション

ポリシーアクションに対するサポート	あり

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件でアクションを実行できるかということです。

JSON ポリシーのAction要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連する AWS API オペレーションと同じです。一致する API オペレーションのない権限のみのアクションなど、いくつかの例外があります。また、ポリシーに複数アクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

このアクションは、関連付けられたオペレーションを実行するための権限を付与するポリシーで使用されます。

AWS Ground Station アクションのリストを確認するには、サービス認可リファレンスの「AWS Ground Station で定義されるアクション」を参照してください。

AWS Ground Station のポリシーアクションは、アクションの前に以下のプレフィックス を使用します。

```
groundstation
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [
      "groundstation:action1",
      "groundstation:action2"
         ]
```

AWS Ground Station アイデンティティベースのポリシーの例を表示するには、「AWS Ground Station のアイデンティティベースのポリシーの例」を参照してください。

AWS Ground Station のポリシーリソース

ポリシーリソースに対するサポート	あり

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件でアクションを実行できるかということです。

JSON ポリシーの Resource 要素は、アクションが適用される 1 つ以上のオブジェクトを指定します。ステートメントには、Resource または NotResource 要素を含める必要があります。ベストプラクティスとして、Amazon リソースネーム (ARN) を使用してリソースを指定します。これは、リソースレベルの権限と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

AWS Ground Station リソースのタイプとその ARN のリストを確認するには、「サービス認可リファレンス」の「AWS Ground Station で定義されるリソース」を参照してください。どのアクションで各リソースの ARN を指定できるかについては、「AWS Ground Station で定義されるアクション」を参照してください。

AWS Ground Station アイデンティティベースのポリシーの例を表示するには、「AWS Ground Station のアイデンティティベースのポリシーの例」を参照してください。

AWS Ground Station 向けのポリシー条件キー

サービス固有のポリシー条件キーのサポート	はい

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効になる条件を指定できます。Condition 要素はオプションです。equal や less than などの条件演算子を使用して条件式を作成することによって、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition 要素を指定するか、1 つの Condition 要素に複数のキーを指定すると、AWS は AND 論理演算子を使用してそれらを評価します。単一の条件キーに複数

の値を指定すると、AWS は OR 論理演算子を使用して条件を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「IAM ポリシー要素: 変数およびタグ」を参照してください。

AWS はグローバル条件キーとサービス固有の条件キーをサポートしています。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の「AWS グローバル条件コンテキストキー」を参照してください。

AWS Ground Station の条件キーのリストを確認するには、サービス認可リファレンスの「AWS Ground Station の条件キー」を参照してください。どのアクションおよびリソースと条件キーを使用できるかについては、「AWS Ground Station で定義されるアクション」を参照してください。

AWS Ground Station アイデンティティベースのポリシーの例を表示するには、AWS Ground Station のアイデンティティベースのポリシーの例を参照してください。

AWS Ground Station の ACL

ACL のサポート	なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための権限を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

AWS Ground Station による ABAC

ABAC のサポート (ポリシー内のタグ)	はい

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義するアクセス許可戦略です。AWS では、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール)、および多数の AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合に操作を許可するように ABAC ポリシーを設計します。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを管理するには、aws:ResourceTag/*key-name*、aws:RequestTag/*key-name*、または aws:TagKeys の条件キーを使用して、ポリシーの Condition 要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーのすべてをサポートする場合、そのサービスでのサポート状況の値は「はい」になります。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、『IAM ユーザーガイド』の「ABAC とは?」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「属性ベースのアクセス制御 (ABAC) を使用する」を参照してください。

AWS Ground Station での一時的な認証情報の使用

一時的な認証情報のサポート	あり

AWS のサービス には、一時的な認証情報を使用してサインインしても機能しないものがあります。一時的な認証情報で機能する AWS のサービスなどの詳細については、「IAM ユーザーガイド」の「IAM と連携する AWS のサービス」を参照してください。

ユーザー名とパスワード以外の方法で AWS Management Console にサインインする場合は、一時的な認証情報を使用していることになります。例えば、会社の Single Sign-On (SSO) リンクを使用して AWS にアクセスすると、そのプロセスは自動的に一時認証情報を作成します。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、『IAM ユーザーガイド』の「ロールへの切り替え (コンソール)」を参照してください。

一時認証情報は、AWS CLI または AWS API を使用して手動で作成できます。作成後、一時的な認証情報を使用して AWS にアクセスできるようになります。AWS は、長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成することをお勧めします。詳細については、「IAM の一時的セキュリティ認証情報」を参照してください。

AWS Ground Station のクロスサービスプリンシパル権限

フォワードアクセスセッション (FAS) をサポート	はい

IAM ユーザーまたはロールを使用して AWSでアクションを実行するユーザーは、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行してから、別のサービスの別のアクションを開始することがあります。FAS は、AWS のサービスを呼び出すプリンシパルの権限を、AWS のサービスのリクエストと合わせて使用し、ダウンストリームのサービスに対してリクエストを行います。FAS リクエストは、サービスが、完了するために他の AWS のサービス またはリソースとのやりとりを必要とするリクエストを受け取ったときにのみ行われます。この場合、両方のアクションを実行するための権限が必要です。FASリクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

AWS Ground Station のサービスロール

サービスロールのサポート	いいえ

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービス にアクセス許可を委任するロールの作成」を参照してください。

> **Warning**
>
> サービスロールのアクセス許可を変更すると、AWS Ground Station の機能が破損する可能性があります。AWS Ground Station が指示する場合以外は、サービスロールを編集しないでください。

AWS Ground Station のサービスリンクロール

サービスリンクロールのサポート	はい

サービスリンクロールは、AWS のサービス にリンクされているサービスロールの一種です。サービスがロールを引き受け、ユーザーに代わってアクションを実行できるようになります。サービスリンクロールは、AWS アカウント に表示され、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールの権限を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「IAM と提携する AWS のサービス」を参照してください。表の中から、「サービスにリンクされたロール」列が「Yes」になって

いるサービスを見つけます。サービスリンクロールに関するドキュメントをサービスで表示するには、「はい」リンクを選択します。

AWS Ground Station のアイデンティティベースのポリシーの例

デフォルトでは、ユーザーおよびロールには、AWS Ground Station リソースを作成または変更する権限はありません。また、AWS Management Console、AWS Command Line Interface (AWS CLI)、または AWS API を使用してタスクを実行することもできません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者がロールに IAM ポリシーを追加すると、ユーザーはロールを引き受けることができます。

これらサンプルの JSON ポリシードキュメントを使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「IAM ポリシーの作成」を参照してください。

AWS Ground Station が定義するアクションとリソースタイプ (リソースタイプごとの ARN のフォーマットを含む) の詳細については、サービス認証リファレンスの「AWS Ground Station のアクション、リソース、および条件キー」を参照してください。

トピック

- ポリシーのベストプラクティス
- AWS Ground Station コンソールを使用する
- 自分の権限の表示をユーザーに許可する

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが AWS Ground Station リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションを実行すると、AWS アカウント に料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS マネージドポリシーを使用して開始し、最小特権の権限に移行する – ユーザーとワークロードへの権限の付与を開始するには、多くの一般的なユースケースのために権限を付与する AWS マネージドポリシーを使用します。これらは AWS アカウントで使用できます。ユースケースに応じた AWS カスタマーマネージドポリシーを定義することで、権限をさらに減らすことをお勧めします。詳細については、『IAM ユーザーガイド』の「AWS マネージドポリシー」または「AWS ジョブ機能の管理ポリシー」を参照してください。

- 最小特権を適用する – IAM ポリシーで権限を設定するときは、タスクの実行に必要な権限のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権権限とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、『IAM ユーザーガイド』の「IAM でのポリシーと権限」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。また、AWS CloudFormation などの特定の AWS のサービス を介して使用する場合、条件を使用してサービスアクションへのアクセスを許可することもできます。詳細については、『IAM ユーザーガイド』の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、『IAM ユーザーガイド』の「IAM Access Analyzer ポリシーの検証」を参照してください。
- 多要素認証 (MFA) を要求する - AWS アカウント内の IAM ユーザーまたはルートユーザーを要求するシナリオがある場合は、セキュリティを強化するために MFA をオンにします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、『IAM ユーザーガイド』の「MFA 保護 API アクセスの設定」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

AWS Ground Station コンソールを使用する

AWS Ground Station コンソールにアクセスするには、一連の最小限のアクセス許可が必要です。これらのアクセス許可により、AWS アカウント の AWS Ground Station リソースの詳細をリストおよび表示できます。最小限の必要なアクセス許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しなくなります。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソール権限を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションへのアクセスのみを許可します。

ユーザーとロールが引き続き AWS Ground Station コンソールを使用できるようにするには、エンティティに AWS Ground Station *ConsoleAccess* または *ReadOnly* AWS 管理ポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「ユーザーへのアクセス許可の追加」を参照してください。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI か AWS API を使用してプログラム的に、このアクションを完了するアクセス許可が含まれています。

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

```
    ]
}
```

AWS Ground Station ID とアクセスのトラブルシューティング

以下の情報は、AWS Ground Station と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

トピック

- AWS Ground Station でアクションを実行する権限がない
- iam:PassRole を実行する権限がない
- 自分の AWS アカウント 以外のユーザーに AWS Ground Station リソースへのアクセスを許可したい

AWS Ground Station でアクションを実行する権限がない

あるアクションを実行する権限がないというエラーが表示された場合、そのアクションを実行できるようにポリシーを更新する必要があります。

次の例は、mateojackson という IAM ユーザーがコンソールを使用して架空の *my-example-widget* リソースに関する詳細を表示しようとしたとき、架空の groundstation:*GetWidget* アクセス許可がない場合に発生するエラーを示しています。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 groundstation:GetWidget on resource: my-example-widget
```

この場合、groundstation:*GetWidget* アクションを使用して *my-example-widget*リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、AWS 管理者に問い合わせてください。管理者とは、サインイン認証情報を提供した担当者です。

iam:PassRole を実行する権限がない

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して AWS Ground Station にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールやサービスリンクロールを作成せずに、既存のロールをサービスに渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して AWS Ground Station でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、Mary のポリシーを更新して、Mary に iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者に問い合わせてください。サインイン認証情報を提供した担当者が管理者です。

自分の AWS アカウント 以外のユーザーに AWS Ground Station リソースへのアクセスを許可したい

他のアカウントのユーザーや組織外のユーザーが、リソースへのアクセスに使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセス制御リスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください。

- AWS Ground Station がこれらの機能をサポートしているかどうかを確認するには、「AWS Ground Station と IAM の連携方法」をご参照ください。
- 所有している AWS アカウント 全体のリソースへのアクセス権を付与する方法については、「IAM ユーザーガイド」の「所有している別の AWS アカウント アカウントへのアクセス権を IAM ユーザーに付与する」を参照してください。
- サードパーティーの AWS アカウント にリソースへのアクセス権を提供する方法については、『IAM ユーザーガイド』の「第三者が所有する AWS アカウント へのアクセス権を付与する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、『IAM ユーザーガイド』の「外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可」を参照してください。

- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、『IAM ユーザーガイド』の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

Ground Station のサービスにリンクされたロールの使用

AWS Ground Station は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスにリンクされたロールは、Ground Station に直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは、Ground Station によって事前定義されており、ユーザーの代わりにサービスから他の AWS のサービスを呼び出す必要のあるアクセス許可がすべて含まれています。

サービスにリンクされたロールを使用することで、必要なアクセス許可を手動で追加する必要がなくなるため、Ground Station の設定が簡単になります。Ground Station は、サービスにリンクされたロールのアクセス許可を定義し、他の定義がされている場合を除き、Ground Station のみがそのロールを引き受けることができます。定義した許可には、信頼ポリシーと許可ポリシーが含まれます。この許可ポリシーを他のIAM エンティティにアタッチすることはできません。

サービスにリンクされたロールをサポートする他のサービスについては、「IAM と連動する AWS のサービス」を参照し、[Service-linked roles] (サービスにリンクされたロール) の列内で [Yes] (はい) と表記されたサービスを確認してください。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Ground Station のサービスにリンクされたロールのアクセス許可

Ground Station は、AWSServiceRoleForGroundStationDataflowEndpointGroup というサービスにリンクされたロールを使用します – AWS GroundStation は、このサービスにリンクされたロールを使用して EC2 を呼び出し、パブリック IPv4 アドレスを検索します。

AWSServiceRoleForApplicationDiscoveryServiceContinuousExport という、サービスにリンクされたロールは、以下のサービスを信頼してロールを引き受けます。

- `groundstation.amazonaws.com`

AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy というロールのアクセス許可ポリシーでは、Ground Station は、指定されたリソースで次のアクションを完了することができます。

- アクション: `all AWS resources (*)` 上で `ec2:DescribeAddresses`

アクションにより、Ground Station は EIP に関連付けられているすべての IP を一覧表示できます。

- アクション: `all AWS resources (*)` 上で `ec2:DescribeNetworkInterfaces`

  アクションにより、Ground Station は EC2 インスタンスに関連付けられたネットワークインターフェイスに関する情報を取得できます

サービスにリンクされたロールの作成、編集、削除をIAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、許可を設定する必要があります。詳細については、「IAM User Guide」(IAM ユーザーガイド) の「Service-linked role permissions」(サービスにリンクされたロールのアクセス権限) を参照してください。

Ground Station へのサービスにリンクされたロールの作成

サービスにリンクされたロールを手動で作成する必要はありません。AWS CLI または AWS API で DataflowEndpointGroup を作成すると、Ground Station によってサービスにリンクされたロールが作成されます。

このサービスにリンクされたロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。DataflowEndpointGroup を作成すると、Ground Station によってサービスにリンクされたロールが再作成されます。

IAM コンソールを使用して、Amazon EC2 へのデータ配信ユースケースで、サービスにリンクされたロールを作成することもできます。AWS CLI または AWS API で、`groundstation.amazonaws.com` サービス名を使用してサービスリンクロールを作成します。詳細については、IAM ユーザーガイドの「サービスリンクロールの作成」を参照してください。このサービスリンクロールを削除する場合、この同じプロセスを使用して、もう一度ロールを作成できます。

Ground Station でのサービスにリンクされたロールの編集

Ground Station では、サービスにリンクされたロールである AWSServiceRoleForGroundStationDataflowEndpointGroup を編集できません。サービスにリンクされたロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロールの説明の編集はできます。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの編集」を参照してください。

Ground Station でのサービスにリンクされたロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。

サービスにリンクされたロールは、サービスにリンクされたロールを使用して DataflowEndpointGroups を削除した後でしか削除できません。これにより、DataflowEndpointGroups に対するアクセス許可を誤って取り消すことがなくなります。サービスにリンクされたロールが複数の DataflowEndpointGroups で使用されている場合、サービスにリンクされたロールを削除する前に、そのロールを使用するすべての DataflowEndpointGroups を削除する必要があります。

> **Note**
>
> リソースを削除する際に、Ground Station のサービスでそのロールが使用されている場合、削除は失敗することがあります。失敗した場合は、数分待ってから操作を再試行してください。

AWSServiceRoleForGroundStationDataflowEndpointGroup によって使用される Ground Station リソースを削除するには

- AWS CLI または AWS API を使用して DataflowEndpointGroups を削除します。

IAM を使用してサービスリンクロールを手動で削除するには

IAM コンソール、AWS CLI、または AWS API を使用して、AWSServiceRoleForGroundStationDataflowEndpointGroup というサービスにリンクされたロールを削除します。詳細については、IAM ユーザーガイドの「サービスにリンクされたロールの削除」を参照してください。

Ground Station のサービスにリンクされたロールがサポートされるリージョン

Ground Station は、サービスが利用可能なすべてのリージョンで、サービスにリンクされたロールの使用をサポートします。詳細については、「リージョン表」を参照してください。

トラブルシューティング

NOT_AUTHORIZED_TO_CREATE_SLR - これは、CreateDataflowEndpointGroup API の呼び出しに使用されているアカウントのロールに iam:CreateServiceLinkedRole のアクセス許可がないことを示しています。iam:CreateServiceLinkedRole のアクセス許可を持つ管理者は、アカウントのサービスにリンクされたロールを手動で作成する必要があります。

AWS の AWS Ground Station マネージドポリシー

AWS マネージドポリシーは、AWS が作成および管理するスタンドアロンポリシーです。AWS マネージドポリシーは、多くの一般的なユースケースでアクセス許可を提供できるように設計されているため、ユーザー、グループ、ロールへのアクセス許可の割り当てを開始できます。

AWS マネージドポリシーは、ご利用の特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることにご注意ください。AWS のすべてのお客様が使用できるようになるのを避けるためです。ユースケース別にカスタマーマネージドポリシーを定義することで、アクセス許可を絞り込むことをお勧めします。

AWS マネージドポリシーで定義したアクセス権限は変更できません。AWS が AWS マネージドポリシーに定義されているアクセス許可を更新すると、更新はポリシーがアタッチされているすべてのプリンシパルアイデンティティ (ユーザー、グループ、ロール) に影響します。新しい AWS のサービスを起動するか、既存のサービスで新しい API オペレーションが使用可能になると、AWS が AWS マネージドポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

AWS マネージドポリシー: AWSGroundStationAgentInstancePolicy

AWSGroundStationAgentInstancePolicy ポリシーは IAM ID にアタッチできます。

このポリシーは、Ground Station コンタクト中にインスタンスがデータを送受信できるようにする、AWS Ground Station エージェントのアクセス許可をお客様のインスタンスに付与します。このポリシーのすべてのアクセス許可は、Ground Station サービスからのものです。

許可の詳細

このポリシーには、以下の許可が含まれています。

- groundstation — データフローエンドポイントインスタンスが Ground Station Agent API を呼び出すことを許可します。

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

AWS マネージドポリシー: AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy

ユーザーの IAM エンティティに、AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy をアタッチすることはできません。このポリシーは、ユーザーに代わって AWS Ground Station がアクションを実行することを許可する、サービスにリンクされたロールにアタッチされます。詳細については、「サービスにリンクされたロールの使用」を参照してください。

このポリシーは、AWS Ground Station がパブリック IPv4 アドレスを検索できるアクセス許可を EC2 に付与します。

許可の詳細

このポリシーには、以下の許可が含まれています。

- `ec2:DescribeAddresses` — AWS Ground Station が EIP に関連付けられているすべての IP をユーザーに代わって一覧表示できます。
- `ec2:DescribeNetworkInterfaces` — AWS Ground Station が EC2 インスタンスに関連するネットワークインターフェイスに関する情報をユーザーに代わって取得できます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Ground Station マネージドポリシーの AWS 更新

このサービスがこれらの変更の追跡を開始してからの、AWS の AWS Ground Station マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動通知については、AWS Ground Station [Document history] (ドキュメントの履歴) ページの RSS フィードをサブスクライブしてください。

変更	説明	日付
AWSGroundStationAgentInstancePolicy — 新しいポリシー	AWS Ground Station では、データフローエンドポイントインスタンスに AWS Ground	2023 年 4 月 12 日

変更	説明	日付
	Station Agent を使用するためのアクセス許可を提供する新しいポリシーを追加しました。	
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy — 新しいポリシー	AWS Ground Station では、EIP に関連付けられたパブリック IPv4 アドレスと EC2 インスタンスに関連付けられたネットワークインターフェイスを AWS Ground Station が検索できるようにするアクセス許可を EC2 に付与する新しいポリシーを追加しました。	2022 年 11 月 2 日
AWS Ground Station は変更の追跡を開始しました	AWS Ground Station が AWS マネージドポリシーの変更の追跡を開始しました。	2021 年 3 月 1 日

保存時のデータ暗号化 AWS Ground Station

AWS Ground Station 保存中のお客様の機密データを独自の暗号化キーを使用して保護するため、AWS デフォルトで暗号化を行います。

- AWS 所有のキー- AWS Ground Station デフォルトでこれらのキーを使用して、直接識別可能な個人データや工フェメリドを自動的に暗号化します。AWS が所有するキーを表示、管理、使用したり、その使用状況を監査したりすることはできません。ただし、データを暗号化するキーを保護するためのアクションの実行や、プログラムの変更は必要ありません。詳細については、「AWS Key Management Service デベロッパーツールガイド」の「AWS が所有するキー」を参照してください。

保管中のデータをデフォルトで暗号化することで、機密データの保護におけるオーバーヘッドと複雑な作業を減らすのに役立ちます。同時に、セキュリティを重視したアプリケーションを構築して、暗号化のコンプライアンスと規制の厳格な要件を満たすことができます。

AWS Ground Station 保存中の機密データはすべて暗号化されますが、 AWS Ground Station 工フェメリドなどの一部のリソースでは、デフォルトの管理キーの代わりにカスタマー管理キーを使用することもできます。 AWS

- 顧客管理鍵--ユーザーが作成、所有、 AWS Ground Station 管理する対称の顧客管理鍵の使用をサポートします。これにより、既存の所有暗号化にさらに暗号化のレイヤーが追加されます。 AWS この暗号化層はユーザーが完全に制御できるため、次のようなタスクを実行できます。
  - キーポリシーの策定と維持
  - IAM ポリシーとグラントの策定と維持
  - キーポリシーの有効化と無効化
  - 暗号化素材のローテーション
  - タグの追加
  - キーエイリアスの作成
  - キー削除のスケジュール設定

  詳細については、「AWS Key Management Service デベロッパーガイド」の「カスタマーマネージドキー」を参照してください。

次の表は、 AWS Ground Station カスタマー管理キーの使用をサポートするリソースをまとめたものです。

データタイプ	AWS が所有するキーの暗号化	カスタマーマネージドキーの暗号化 (オプション)
衛星の軌跡の計算に使用されるエフェメリスデータ	有効	有効

> **Note**
>
> AWS Ground Station 個人を特定できるデータを無料で保護するために、 AWS 所有しているキーを使用して保管中の暗号化を自動的に有効にします。ただし、お客様が管理するキーの使用には AWS KMS の料金がかかります。料金の詳細については、「AWS Key Management Service の料金表」を参照してください。
> KMS の詳細については、『AWS AWS KMS 開発者ガイド』を参照してください。

KMS AWS Ground Station での権限の使用方法 AWS

AWS Ground Station カスタマーマネージドキーを使用するにはキーグラントが必要です。

カスタマー管理キーで暗号化されたエフェメリスをアップロードすると、KMS にリクエストを送信して、 AWS Ground Station ユーザーに代わってキーグラントを作成します。 CreateGrant AWS AWS KMS の権限は、顧客アカウントの KMS AWS Ground Station キーへのアクセス権を付与するために使用されます。

AWS Ground Station お客様の顧客管理キーを以下の内部操作に使用するように権限を付与する必要があります。

- AWS KMS `GenerateDataKey` にリクエストを送信して、カスタマーマネージドキーで暗号化されたデータキーを生成してください。
- AWS KMS `Decrypt` にリクエストを送信して、暗号化されたデータキーを復号化して、データの暗号化に使用できるようにします。
- AWS KMS `Encrypt` にリクエストを送信して、提供されたデータを暗号化してください。

任意のタイミングで、許可に対するアクセス権を取り消したり、カスタマーマネージドキーに対するサービスからのアクセス権を削除したりできます。そうすると、 AWS Ground Station カスタマー管理キーによって暗号化されたデータにはアクセスできなくなり、そのデータに依存する操作に影響します。たとえば、連絡先に現在使用されているエフェメリスからキーグラントを削除すると、 AWS Ground Station 提供されたエフェメリスデータを使用して連絡中にアンテナを向けることができなくなります。これにより、コンタクトは FAILED 状態で終了します。

カスタマーマネージドキーを作成する

管理コンソールまたは KMS API を使用して、対称型の顧客管理キーを作成できます。 AWS AWS

対称カスタマーマネージドキーを作成するには

『キー管理サービス開発者ガイド』の「対称顧客管理キーの作成」 AWS の手順に従ってください。

キーポリシー

キーポリシーは、カスタマーマネージドキーへのアクセスを制御します。すべてのカスタマーマネージドキーには、キーポリシーが 1 つだけ必要です。このポリシーには、そのキーを使用できるユーザーとその使用方法を決定するステートメントが含まれています。カスタマーマネージドキーを作成する際に、キーポリシーを指定することができます。詳細については、『Key Management Service 開発者ガイド』の「AWS カスタマー管理キーへのアクセスの管理」を参照してください。

AWS Ground Station カスタマー管理キーをリソースで使用するには、キーポリシーで次の API 操作を許可する必要があります。

`kms:CreateGrant` - カスタマーマネージドキーに許可を追加します。指定した KMS キーへのアクセスを許可します。これにより、AWS Ground Station 必要な権限付与操作へのアクセスが可能になります。権限の使用について詳しくは、『 AWS キー管理サービス開発者ガイド』を参照してください。

これにより、Amazon AWS では次のことが可能になります。

- `GenerateDataKey` を呼び出して、暗号化されたデータキーを生成して保存します。データキーは暗号化にすぐには使用されないからです。
- `Decrypt` を呼び出して、保存された暗号化データキーを使用して暗号化されたデータにアクセスします。
- `Encrypt` を呼び出して、データキーを使用してデータを暗号化します。
- `RetireGrant` にサービスが許可するための、廃止するプリンシパルを設定します。

kms:DescribeKey-お客様が管理するキーの詳細を提供することで AWS Ground Station 、提供されたキーに対してグラントを作成する前にキーを検証できます。

追加できる IAM ポリシーステートメントの例を以下に示します。 AWS Ground Station

```
"Statement" : [
  {"Sid" : "Allow access to principals authorized to use AWS Ground Station",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:DescribeKey",
      "kms:CreateGrant"
    ],
    "Resource" : "*",
    "Condition" : {
    "StringEquals" : {
        "kms:ViaService" : "groundstation.amazonaws.com",
        "kms:CallerAccount" : "111122223333"
    }
  },
  {"Sid": "Allow access for key administrators",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action" : [
      "kms:*"
      ],
    "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
  },
  {"Sid" : "Allow read-only access to key metadata to the account",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : [
      "kms:Describe*",
      "kms:Get*",
      "kms:List*",
      "kms:RevokeGrant"
    ],
    "Resource" : "*"
```

```
  }
]
```

ポリシーでのアクセス権限の指定について詳しくは、『 AWS キー管理サービス開発者ガイド』を参照してください。

キーアクセスのトラブルシューティングについて詳しくは、『 AWS キー管理サービス開発者ガイド』を参照してください。

カスタマー管理キーの指定 AWS Ground Station

カスタマーマネージドキーを指定して、次のリソースを暗号化できます。

- エフェメリス

リソースを作成するときに、以下のように指定することでデータキーを指定できます。kmsKeyArn

- kmsKeyArn- AWS KMS カスタマー管理キーのキー識別子

AWS Ground Station 暗号化コンテキスト

暗号化コンテキストは、データに関する追加のコンテキスト情報が含まれたキーバリューペアのオプションのセットです。 AWS KMS は、認証済み暗号化をサポートするための追加の認証データとして暗号化コンテキストを使用します。データを暗号化するリクエストに暗号化コンテキストを含めると、 AWS KMS は暗号化コンテキストを暗号化されたデータにバインドします。データを復号化するには、そのリクエストに (暗号化時と) 同じ暗号化コンテキストを含めます。

AWS Ground Station 暗号化コンテキスト

AWS Ground Station 暗号化されるリソースに応じて異なる暗号化コンテキストを使用し、作成されるキーグラントごとに特定の暗号化コンテキストを指定します。

エフェメリス暗号化コンテキスト:

エフェメリスリソースを暗号化するためのキー許可は、特定の衛星 ARN にバインドされます。

```
"encryptionContext": {
    "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
```

```
}
```

> **Note**
>
> キーグ許可は同じキーと衛星のペアに再利用されます。

暗号化コンテキストによるモニタリングに暗号化コンテキストを使用する

対称カスタマーマネージドキーを使用してエメリフィスを暗号化する場合は、監査レコードとログで暗号化コンテキストを使用して、カスタマーマネージドキーがどのように使用されているかを特定することもできます。暗号化コンテキストは、AWS CloudTrail または Amazon Logs CloudWatch によって生成されたログにも表示されます。

暗号化コンテキストを使用してカスタマーマネージドキーへのアクセスを制御する

対称カスタマーマネージドキー (CMK) へのアクセスを制御するための conditions として、キーポリシーと IAM ポリシー内の暗号化コンテキストを使用することができます。付与する際に、暗号化コンテキストの制約を使用することもできます。

AWS Ground Station グラントの暗号化コンテキスト制約を使用して、アカウントまたはリージョンのカスタマーマネージドキーへのアクセスを制御します。権限の制約では、権限によって許可されるオペレーションで指定された暗号化コンテキストを使用する必要があります。

次に、特定の暗号化コンテキストのカスタマーマネージドキーへのアクセスを付与するキーポリシーステートメントの例を示します。このポリシーステートメントの条件では、権限に暗号化コンテキストを指定する暗号化コンテキスト制約が必要です。

```
{"Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": "kms:DescribeKey",
     "Resource": "*"
},{"Sid": "Enable CreateGrant",
     "Effect": "Allow",
     "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
```

```
      },
      "Action": "kms:CreateGrant",
      "Resource": "*",
      "Condition": {
         "StringEquals": {
             "kms:EncryptionContext:aws:groundstation:arn":
  "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
         }
      }
}
```

以下の暗号化キーを監視します。 AWS Ground Station

AWS KMS AWS Ground Station カスタマー管理キーをリソースで使用すると、AWS CloudTrailまたは Amazon CloudWatch ログを使用して AWS KMS AWS Ground Station に送信されるリクエストを追跡できます。次の例はCreateGrant、、、GenerateDataKeyDecrypt、EncryptDescribeKey AWS カスタマーマネージドキーで暗号化されたデータにアクセスするためにGGround Station によって呼び出される KMS AWS CloudTrail 操作を監視するためのイベントです。

CreateGrant (Cloudtrail)

AWS KMS カスタマー管理キーを使用してエフェメリスリソースを暗号化すると、ユーザーに代わってアカウントの KMS AWS Ground Station CreateGrant キーへのアクセスリクエストが送信されます。 AWS AWS Ground Station 作成される権限は、KMS カスタマー管理キーに関連付けられたリソースに固有のものです。 AWS さらに、リソースを削除すると、 AWS GGround Station RetireGrant はこの操作を使用してグラントを削除します。

以下のイベント例では CreateGrant オペレーションを記録しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AAAAAAAAAAAAAAAAAAAAA:SampleUser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE3",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
```

```
                "principalId": "AAAAAAAAAAAAAAAAAAAAA",
                "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": "111.11.11.11",
    "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 (Cloudtrail)

AWS KMS カスタマー管理キーを使用してエフェメリスリソースを暗号化すると、 AWS Ground Station DescribeKeyリクエストされたキーがアカウントに存在することを確認するリクエストがユーザーに代わって送信されます。

以下のイベント例では DescribeKey オペレーションを記録しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AAAAAAAAAAAAAAAAAAAAA:SampleUser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/User/Role",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE3",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AAAAAAAAAAAAAAAAAAAAA",
                "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 (Cloudtrail)

AWS KMS カスタマー管理キーを使用して エフェメリスリソースを暗号化すると、GenerateDataKeyデータを暗号化するためのデータキーを生成するリクエストが KMS AWS Ground Station に送信されます。

以下のイベント例では GenerateDataKey オペレーションを記録しています。

```
{
    "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 (Cloudtrail)

AWS KMS カスタマー管理キーを使用してエフェメリスリソースを暗号化すると、提供されたエフェメリスが同じカスタマー管理キーですでに暗号化されている場合は、 AWS Ground Station Decryptオペレーションを使用してそのエフェメリスを復号化します。例えば、エフェメリスが S3 バケットからアップロードされ、そのバケット内で特定のキーで暗号化されているとします。

以下のイベント例では Decrypt オペレーションを記録しています。


```
{
    "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"
}
```

衛星エフェメリスデータ

エフェメリス (単数形: ephemeris、複数形: ephemerides) は、天体の軌道を提供するファイルまたはデータ構造です。従来、このファイルは表形式のデータのみを参照していましたが、次第に、宇宙機の軌道を示すさまざまなデータファイルを参照するようになりました。

AWS Ground Station エフェメリスデータを使用して、衛星の連絡先がいつ利用可能になるかを判断し、 AWS Ground Station ネットワーク内のアンテナに衛星を指すように正しく命令します。デフォルトでは、エフェメリドを提供するためのアクションは必要ありません。 AWS Ground Station

トピック

- デフォルトのエフェメリスデータ
- どのエフェメリスが使われているか
- 衛星用の現在のエフェメリスの取得
- カスタムエフェメリスデータの提供
- 無効なエフェメリスのトラブルシューティング
- デフォルトのエフェメリスデータに戻す

デフォルトのエフェメリスデータ

デフォルトでは、 AWS Ground Station は Space-Track から公開されているデータを使用するため、 AWS Ground Station これらのデフォルトエフェメリドを提供するためのアクションは必要ありません。これらのエフェメリスは、衛星の NORAD ID に関連付けられた 2 行軌道要素セットです。すべてのデフォルトのエフェメリスの優先度は 0 です。そのため、デフォルトのエフェメリスは、エフェメリス API 経由でアップロードされた有効期限が切れていないカスタムエフェメリスがあれば、それによって常に上書きされます。このカスタムエフェメリスは、常に優先度が 1 以上である必要があります。

NORAD ID のないサテライトは、カスタムエフェメリスデータをにアップロードする必要があります。 AWS Ground Station例えば、打ち上げたばかりの衛星や Space-Track カタログから意図的に除外された衛星には NORAD ID がないため、カスタムエフェメリスをアップロードする必要があります。カスタムエフェメリスの提供に関する詳細は、「カスタムエフェメリスデータの提供」を参照してください。

どのエフェメリスが使われているか

エフェメリスには、優先度、有効期限、有効フラグがあります。これらを総合して、どのエフェメリスを衛星に使うかが決まります。1 つの衛星でアクティブにできるエフェメリスは 1 つだけです。

使用されるエフェメリスは、有効期限が今後のもので優先度が最も高く、有効になっているエフェメリスです。によって返される連絡可能時間はListContacts、このエフェメリスに基づいています。複数の ENABLED エフェメリスの優先度が同じ場合は、最後に作成または更新されたエフェメリスが使用されます。

> **Note**
>
> AWS Ground Station ENABLEDサテライトあたりの顧客提供のエフェメリドの数にサービスクォータがあります (「Service Quotas」を参照)。このクォータに達した後にエフェメリスデータをアップロードするには、お客様が提供した最も低い優先度のエフェメリス/最も作成日の古いエフェメリスを削除 (`DeleteEphemeris` を使用) または無効 (`UpdateEphemeris` を使用) にします。

エフェメリスが作成されていない場合、ENABLEDまたはエフェメリドにステータスが設定されていない場合は、(Space Track の) AWS Ground Station サテライト用のデフォルトのエフェメリスが利用可能であれば使用します。このデフォルトのエフェメリスの優先度は 0 です。

新しいエフェメリスが以前にスケジュールされたコンタクトに与える影響

DescribeContact API を使用すると、アクティブな可視時間を返すことで、以前にスケジュールされた連絡先に新しいエフェメリドが及ぼす影響を確認できます。

新しいエフェメリスをアップロードする前にスケジュールされたコンタクトは、当初の予定コンタクト時間を保持しますが、アンテナトラッキングではアクティブなエフェメリスが使用されます。アクティブなエフェメリスに基づく宇宙船の位置が以前のエフェメリスと大きく異なる場合、宇宙船は送信/受信サイトマスクの外で動作するため、衛星とアンテナとの接触時間が短くなる可能性があります。そのため、以前のエフェメリスとは大きく異なる新しいエフェメリスをアップロードした後で、future 連絡先をキャンセルして再スケジュールすることをお勧めします。DescribeContact この API を使用すると、`startTime`と`endTime`スケジュールされた連絡先と返されたおよびを比較することで、送信/受信サイトマスクの外で動作している宇宙船が原因で、future 連絡先が使用できなくなる部分を特定できます。`visibilityStartTime` `visibilityEndTime`future 連絡をキャンセルして再スケジュールする場合は、連絡時間範囲が可視時間範囲から30秒以上外れてはいけません。

キャンセルされた連絡は、連絡時刻が近づきすぎると費用が発生する可能性があります。キャンセルされたコンタクトの詳細については、「Ground Station FAQ」を参照してください。

衛星用の現在のエフェメリスの取得

AWS Ground Station 特定のサテライトで現在使用されているエフェメリスは、またはアクションを呼び出すことで取得できます。GetSatellite ListSatellitesこれらのメソッドはいずれも、現在使用中のエフェメリスのメタデータを返します。このエフェメリスのメタデータは、アップロードされたカスタムエフェメリドとデフォルトのエフェメリドでは異なります。 AWS Ground Station

デフォルトのエフェメリスには、source と epoch のフィールドのみが含まれます。epochこれは Space Trackから引き出された2行要素集合の時代で AWS Ground Station、現在は衛星の軌道の計算に使われている。

カスタムエフェメリスは、source 値が "CUSTOMER_PROVIDED" となり、ephemerisId フィールドには一意の識別子が含まれます。この一意の識別子を使用して、DescribeEphemeris アクションを通じてエフェメリスをクエリできます。アクションによるアップロード中にエフェメリスに名前が割り当てられた場合は、nameオプションのフィールドが返されます。 AWS Ground Station CreateEphemeris

AWS Ground Station エフェメリドはによって動的に更新されるため、返されるデータは API の呼び出し時に使用されていたエフェメリスのスナップショットにすぎない点に注意することが重要です。

デフォルトのエフェメリスを使用する衛星用の **GetSatellite** の戻り値例

```
{
    "satelliteId": "e1cfe0c7-67f9-4d98-bad2-06dbfc2d14a2",
    "satelliteArn": "arn:aws:groundstation::111122223333:satellite/e1cfe0c7-67f9-4d98-
bad2-06dbfc2d14a2",
    "noradSatelliteID": 12345,
    "groundStations": [
        "Example Ground Station 1",
        "Example Ground Station 2"
    ],
    "currentEphemeris": {
        "source": "SPACE_TRACK",
        "epoch": 8888888888
    }
```

```
}
```

カスタムエフェメリスを使用する衛星用の**GetSatellite** の戻り値の例

```
{
      "satelliteId": "e1cfe0c7-67f9-4d98-bad2-06dbfc2d14a2",
      "satelliteArn": "arn:aws:groundstation::111122223333:satellite/
e1cfe0c7-67f9-4d98-bad2-06dbfc2d14a2",
      "noradSatelliteID": 12345,
      "groundStations": [
          "Example Ground Station 1",
          "Example Ground Station 2"
      ],
      "currentEphemeris": {
          "source": "CUSTOMER_PROVIDED",
          "ephemerisId": "e1cfe0c7-67f9-4d98-bad2-06dbfc2d14a2",
          "name": "My Ephemeris"
      }
  }
```

カスタムエフェメリスデータの提供

> **Warning**
>
> エフェメリス API は現在プレビュー状態です。

エフェメリス API へのアクセスは、必要な場合にのみ提供されます。カスタムエフェメリスデータをアップロードする機能を必要とするお客様は、aws-groundstation@amazon.com までご連絡ください。

概要

Ephemeris API を使用すると、 AWS Ground Station カスタムエフェメリドをアップロードして衛星で使用することができます。これらのエフェメリスは、Space Track からのデフォルトのエフェメリスを上書きします (「デフォルトのエフェメリスデータ」を参照)。

顧客天体暦をアップロードすることで、追跡の質が向上し、スペーストラック天体暦が利用できない初期の運用に対応し、操縦を考慮に入れることができます。 AWS Ground Station

カスタムエフェメリスの作成

AWS Ground Station API の CreateEphemeris アクションを使用してカスタムエフェメリスを作成できます。このアクションによって、リクエスト本文または指定された S3 バケットのデータを使用してエフェメリスがアップロードされます。

エフェメリスがアップロードされると、エフェメリスが VALIDATING に設定されて非同期ワークフローが開始されることに注意してください。このワークフローでは、エフェメリスを検証し、そのエフェメリスを基に潜在的なコンタクトを生成します。エフェメリスは、このワークフローを実施して ENABLED になって初めて、コンタクトに使用されます。エフェメリスのステータスを DescribeEphemeris でポーリングするか、Cloudwatch イベントを使用してエフェメリスのステータス変更を追跡する必要があります。

無効なエフェメリスのトラブルシューティングについては、「無効なエフェメリスのトラブルシューティング」を参照してください。

API を使用して TLE セットエフェメリスを作成する

AWS Ground Station boto3 クライアントを使用すると、2 行要素 (TLE) セットのエフェメリスを呼び出し経由でアップロードできます。 AWS Ground Station CreateEphemerisこのエフェメリスは、衛星用のデフォルトのエフェメリスデータの代わりに使用されます (「デフォルトのエフェメリスデータ」を参照)。

TLE セットは JSON 形式のオブジェクトで、1 つ以上の TLE をつなぎ合わせて連続した軌道を構築します。TLE セット内の TLE は、軌道の構築に使用できる連続したセットを形成する (つまり、TLE セット内の TLE 間に時間的なギャップがない) 必要があります。TLE セットの例を以下に示します。

```
# example_tle_set.json
[
    {
        "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**
>
> TLE セット内の TLE の時間範囲は、有効で連続的な軌道になるためには正確に一致する必要があります。

TLE セットは boto3 クライアント経由で次のようにアップロードできます。 AWS Ground Station

```
tle_ephemeris_id = ground_station_boto3_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)
                }
            }
        ]
      }
    })
```

この呼び出しは、そのエフェメリスを今後参照するために使用できるエフェメリス ID を返します。例えば、上記の呼び出しで渡されたエフェメリス ID を使って、エフェメリスの状態をポーリングできます。

```
client.describe_ephemeris(ephemerisId=tle_ephemeris_id['ephemerisId'])
```

DescribeEphemeris アクションのレスポンスの例は次のとおりです

```
{
  "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}}]"
    }
  }
}
```

アップロードされたエフェメリスのステータスを追跡するために、DescribeEphemeris ルートをポーリングするか、Cloudwatch イベントを使用することをお勧めします。これは、エフェメリスが ENABLED に設定され、コンタクトのスケジュール設定や実行に使用できるようになる前に、非同期検証ワークフローを実施する必要があるためです。

上記の例では、TLE セット内のすべての TLE の NORAD ID、25994 は、Space Track データベースで衛星に割り当てられている NORAD ID と一致する必要があることに注意してください。

S3 バケットからのエフェメリスデータのアップロード

バケットとオブジェクトキーを指すことで、S3 バケットから直接エフェメリスファイルをアップロードすることもできます。 AWS Ground Station ユーザーに代わってオブジェクトを取得します。AWS Ground Station 保存中のデータの暗号化についての詳細は、「AWS Ground Station の保存時のデータ暗号化」を参照してください。

以下は、S3 バケットから OEM エフェメリスファイルをアップロードする例です。

```
s3_oem_ephemeris_id = customer_client.create_ephemeris( name="2022-10-26 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": "ephemeris-bucket-for-testing",
                "key": "test_data.oem",
            }
        }
    })
```

以下は、以前のサンプルコードブロックでアップロードされた OEM エフェメリスに対して呼び出された DescribeEphemeris アクションから返されるデータの例です。

```
{
    "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"
        }
    }
}
```

無効なエフェメリスのトラブルシューティング

カスタムエフェメリスが AWS Ground Station にアップロードされると、ENABLED になる前に非同期検証ワークフローが実施されます。このワークフローは、衛星識別子、メタデータ、および軌道が有効であることを確保するものです。

エフェメリスが検証に失敗すると、DescribeEphemeris は EphemerisInvalidReason が返します。これで、エフェメリスが検証に失敗した理由が把握できます。EphemerisInvalidReason の潜在的な値は次のとおりです。

値	説明	トラブルシューティングとしてのアクション
METADATA_INVALID	入力された衛星 ID などの宇宙機識別子が無効です	エフェメリスデータに含まれている NORAD ID またはその他の識別子を確認します
TIME_RANGE_INVALID	指定されたエフェメリスの開始時間、終了時間、または有効期限が無効です	開始時間が「今すぐ」より前であること (開始時間を数分前に設定することを推奨)、終了時間が開始時間より後であること、終了時間が有効期限より後であることを確認します
TRAJECTORY_INVALID	入力されたエフェメリスが無効な宇宙機の軌道を定義しています	入力された軌道が連続しており、正しい衛星のものであることを確認します。
VALIDATION_ERROR	エフェメリスの検証処理中に内部サービスエラーが発生しました	アップロードを再試行します

INVALID エフェメリスの DescribeEphemeris レスポンスの例は次のとおりです。

```
{
  "creationTime": 1000000000.00,
  "enabled": false,
  "ephemerisId": "d5a8a6ac-8a3a-444e-927e-EXAMPLE1",
  "name": "Example",
  "priority": 2,
  "status": "INVALID",
  "invalidReason": "METADATA_INVALID",
  "suppliedData": {
    "tle": {
      "sourceS3Object": {
        "bucket": "my-s3-bucket",
        "key": "myEphemerisKey",
        "version": "ephemerisVersion"
      }
```

```
    }
  },
}
```

デフォルトのエフェメリスデータに戻す

カスタムエフェメリスデータをアップロードすると、その特定のサテライトで使用されるデフォルトのエフェメリスデータが上書きされます。 AWS Ground Station AWS Ground Station 現在有効で、有効期限が切れていないお客様提供のエフェメリドが使用できなくなるまで、デフォルトのエフェメリスは使用されません。 AWS Ground Station また、現在の顧客提供のエフェメリスの有効期限を過ぎた連絡先は、その有効期限を過ぎても使用可能なデフォルトのエフェメリスが存在していても一覧表示しません。

デフォルトの Space Track エフェメリスに戻すには、次のいずれかを実行する必要があります。

- 有効になっているお客様提供のエフェメリスをすべて、(`DeleteEphemeris` を使用して) 削除するか、(`UpdateEphemeris` を使用して) 無効にします。衛星用のお客様提供のエフェメリスは、`ListEphemerides` を使用して一覧表示できます。
- 既存のお客様提供のエフェメリスがすべて期限切れになるまで待ちます。

デフォルトのエフェメリスが使用されているかどうかについては、`GetSatellite` を呼び出して、衛星用の現在のエフェメリスの `source` が `SPACE_TRACK` であることを検証することで確認できます。デフォルトのエフェメリスの詳細については、「デフォルトのエフェメリスデータ」を参照してください。

AWS Ground Station サイトマスク

AWS Ground Station 各アンテナロケーションにはサイトマスクが関連付けられています。これらのマスクは、その場所のアンテナが特定の方向 (通常は地平線に近い) を指しているときに、送信または受信をブロックします。マスクには以下の事項が考慮されます。

- アンテナ付近の地理的地形の特徴。例えば、山や建物など、無線周波数 (RF) 信号を遮断したり、送信を妨げたりするものが含まれます。
- 無線周波数干渉 (RFI) これは、受信 (外部 RFI ソースが AWS Ground Station アンテナへのダウンリンク信号に影響を与えるもの) と送信 (AWS Ground Station アンテナによって送信される RF 信号が外部の受信機に悪影響を及ぼすもの) の両方に影響します。
- 法的許可。各リージョンで AWS Ground Station を運用するためのローカルサイトの許可には、送信の最小仰角など、特定の制限が含まれる場合があります。

これらのサイトマスクは時間の経過とともに変化する可能性があります。例えば、アンテナの場所の付近に新しい建物が建設されたり、RFI ソースが変更されたり、法的許可が更新されて異なる制限が適用される場合があります。AWS Ground Station サイトマスクは、秘密保持契約 (NDA) に基づいてお客様に提供されます。

お客様固有のマスク

各サイトの AWS Ground Station サイトマスクに加えて、特定のリージョンの衛星との通信に関する法的許可の制限により、それぞれのお客様に追加のマスクが付与される場合があります。このようなマスクは、AWS Ground Station を使用してこれらの衛星と通信する際のコンプライアンスを確保するために、 case-by-case 基本的に AWS Ground Station で設定することができます。詳細については、AWS Ground Station チームにお問い合わせください。

サイトマスクが利用可能なコンタクト時間に与える影響

サイトマスクには、アップリンク (送信) サイトマスクとダウンリンク (受信) サイトマスクの 2 種類があります。

ListContacts オペレーションを使用して利用可能な接触時間を一覧表示すると、AWS Ground Station は、衛星がダウンリンクマスクより上に上昇し、下に設定される時間に基づいて可視時間を返します。利用可能な連絡時間は、このダウンリンクマスクの可視性ウィンドウに基づいています。

これにより、衛星がダウンリンクマスクを下回っている時間に対して、お客様が予約や料金支払いを行う必要がなくなります。

ミッションプロファイル内のデータフローエッジに Antenna Uplink Config が含まれていても、アップリンクサイトマスクは利用可能なコンタクト時間には適用されません。これにより、アップリンクサイトマスクが原因でアップリンクが利用できない時間でも、お客様は利用可能なすべてのコンタクト時間をダウンリンクに充てることができます。ただし、アップリンク信号は、衛星コンタクト用に予約された時間の一部または全部の間は送信されない場合があります。アップリンク送信をスケジュールする際に、提供されたアップリンクマスクを考慮する責任がお客様にあります。

コンタクトのうち、アップリンクに使用できない部分は、アンテナの場所のアップリンクサイトマスクを基準としたコンタクト中の衛星軌道によって異なります。アップリンクサイトマスクとダウンリンクサイトマスクが類似しているリージョンでは、通常、この期間は短くなります。それ以外の、アップリンクマスクがダウンリンクサイトマスクよりもかなり長いリージョンでは、コンタクト時間の相当部分または全部がアップリンク用として利用できなくなる可能性があります。予約された時間の一部がアップリンクに使用できない場合でも、コンタクト時間の全額がお客様に請求されます。

AWS Ground Station ユーザーガイドのドキュメント履歴

次の表は、 AWS Ground Stationの前回のリリース以降に行われたマニュアルの重要な変更点をまとめたものです。

変更	説明	リリース日
新機能	連絡先は、可視時間範囲から最大 30 秒以内にスケジュールできるようになりました。DescribeContact 可視化時間は回答に含まれます。	2024年3月26日
ドキュメントの更新	組織を改善し、「EC2 インスタンスの選択と CPU 計画」セクションを追加しました。	2024 年 3 月 6 日
ドキュメントの更新	AWS Ground Station サービスとプロセスをエージェントと一緒に実行するための新しいベストプラクティスをエージェントユーザーガイドに追加しました。 AWS Ground Station	2024 年 2 月 23 日
ドキュメントの更新	エージェントリリースノートページを追加しました。	2024 年 2 月 21 日
テンプレートアップデート	DirectBroadcastSatelliteWbDigIfEc2 DataDelivery テンプレートに個別のパブリックサブネットのサポートを追加しました。	2024 年 2 月 14 日
ドキュメントの更新	User Notifications モニタリングドキュメントに AWS への参照を追加しました。	2023 年 8 月 6 日
ドキュメントの更新	コンソールに表示される名前でサテライトにタグを付ける手順を追加しました。 AWS Ground Station	2023 年 7 月 26 日
新機能	ワイドバンド DigIF AWS Ground Station データ配信リリースのエージェントユーザーガイドを追加しました	2023 年 4 月 12 日

変更	説明	リリース日
AWS 管理ポリシーの更新 — AWS 新しい管理ポリシー	AWS Ground Station という名前の新しいポリシーを追加しました AWSGroundStationAgentInstancePolicy。	2023 年 4 月 12 日
新機能	CPE プレビューのリリースに関するユーザーガイドを更新しました。	2022 年 11 月 9 日
AWS 管理ポリシーの更新 — AWS 新しい管理ポリシー	AWS Ground Station AWSServiceRoleForGroundStationDataflowEndpointGroupPolicyという名前の新しいポリシーを含む AWSServiceRoleForGroundStationDataflowEndpointGroup service-linked-role (SLR) を追加しました。	2022 年 11 月 2 日
新機能	AWS CLIとの統合を含むようにユーザーガイドを更新しました。	2020 年 4 月 17 日
新機能	CloudWatch Metrics との統合を含むようにユーザーガイドを更新しました。	2020 年 2 月 24 日
新しいテンプレート	パブリックブロードキャストサテライト (AquaSnppJpss テンプレート) AWS Ground Station がユーザーガイドに追加されました。	2020 年 2 月 19 日
新機能	ユーザーガイドを更新してクロスリージョンのデータ配信を含めました	2020 年 2 月 5 日
ドキュメントの更新	AWS Ground Station CloudWatch イベントを使ったモニタリングの例と説明を更新しました。	2020 年 2 月 4 日
ドキュメントの更新	テンプレートの場所が更新され、「開始方法」セクションと「トラブルシューティング」セクションが改訂されました。	2019 年 12 月 19 日
トラブルシューティングセクションの新規追加	トラブルシューティングセクションが AWS Ground Station ユーザーガイドに追加されました。	2019 年 11 月 7 日

変更	説明	リリース日
新しい「開始方法」トピックの追加	AWS CloudFormation 最新のテンプレートを含む「はじめに」トピックを更新しました。	2019 年 7 月 1 日
Kindle バージョン	AWS Ground Station ユーザーガイドの Kindle バージョンを公開しました。	2019年6月20日
新しいサービスとガイド	AWS Ground Station AWS Ground Station これはユーザーガイドの初回リリースです。	2019 年 5 月 23 日

AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「AWS 用語集」を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。