aws

用户指南

AWS Ground Station

AWS Ground Station: 用户指南

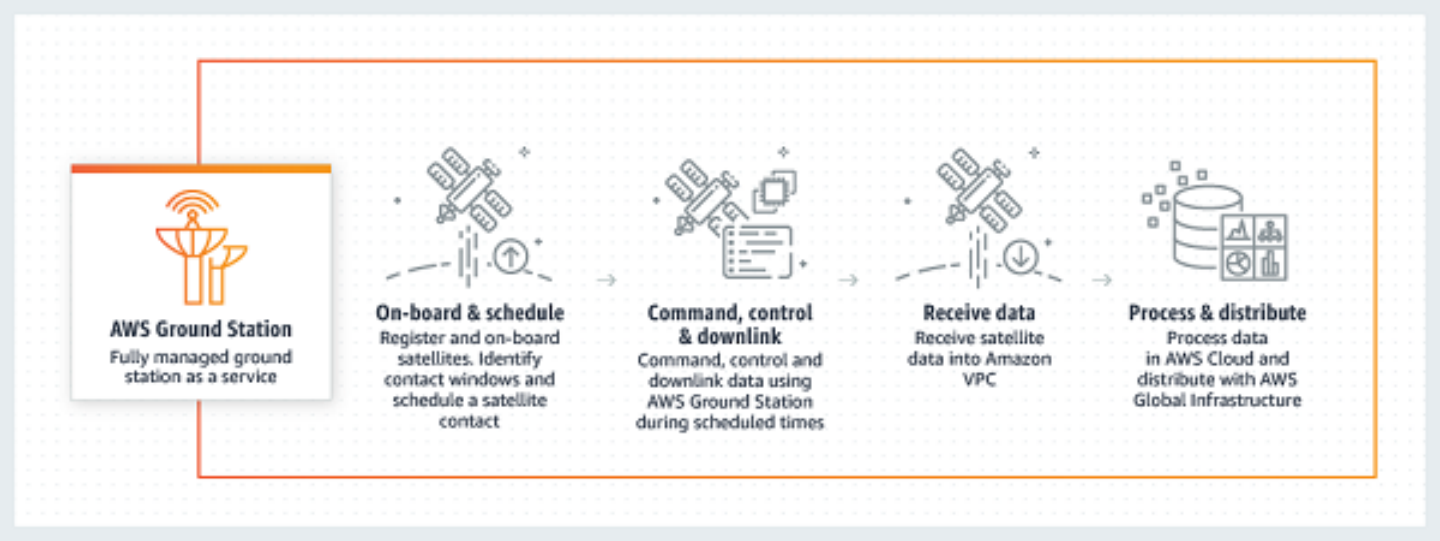
Table of Contents

什么是 AWS Ground Station？

AWS Ground Station 是一项完全托管的服务，可在全球基础设施中提供安全、快速和可预测的卫星通信。有了 AWS Ground Station它，您就不必再构建、管理或扩展自己的地面站基础设施。 AWS Ground Station 使您能够专注于创新和快速尝试采集卫星数据的新应用程序，而不必将资源花在建造、运营和扩展自己的地面站上。

使用 AWS 的低延迟、高带宽的全球光纤网络，您可以在天线系统接收后的几秒钟内开始处理您的卫星数据。这使您能够在几秒钟内将原始数据转化为经过处理的信息或经过分析的知识。

常见使用案例



AWS Ground Station 允许您与卫星进行双向通信，并支持以下用例：

- 下行链路数据 — 接收来自卫星的数据，传输 X 波段和 S 波段频率，实时传输到亚马逊 EC2 实例（VITA-49 格式），或直接发送到您账户中的 Amazon S3 存储桶（PC AP 格式）。此外，对于使用支持的调制和编码方案的卫星，您可以在接收解调和解码的数据或原始数字中频 (digiF) 样本（VITA-49 格式）之间进行选择。
- 上行链路数据 — 通过发送要传输的 digiF 数据（VITA-49 格式），向接收 S 波段频率的卫星发送数据和命令。 AWS Ground Station
- Uplink echo — 通过在物理共置的天线上接收传输的信号，验证发送到航天器的命令，并执行其他高级任务。

- 软件定义无线电 (SDR) /前端处理器 (FEP) — 使用现有 SDR and/or FEP, that's capable of running on an Amazon EC2 instance, to process your data in real-time to send/receive 和现有波形，生成数据产品。
- 遥测、跟踪和指挥 (TT&C)-使用先前列出的用例组合执行 TT&C 来管理您的卫星舰队。
- 跨区域数据传输 — 使用来自单个 AWS 区域 AWS Ground Station的全球天线网络同时操作多个联系人。
- 数字双胞胎 — 无需使用生产天线容量，即可以更低的成本进行测试调度、验证配置和适当的错误处理。

后续步骤

我们建议您首先阅读以下部分：

- 要学习基本 AWS Ground Station 概念，请参阅如何 AWS Ground Station 运作。
- 要了解如何设置账户和要使用的资源 AWS Ground Station，请参阅开始使用。
- 要以编程方式使用 AWS Ground Station，请参阅 AWS Ground Station API 参考。API 参考详细描述了所有 API 操作。 AWS Ground Station 它还提供了受支持的 Web 服务协议的请求、响应和错误示例。您可以使用所选语言的 AWS CLI 或 AWS SDK 来编写与 AWS Ground Station之交互的代码。

如何 AWS Ground Station 运作

AWS Ground Station 操作地面天线以促进与您的卫星的通信。天线能做的事情的物理特征是抽象的，被称为能力。可以在本AWS Ground Station 地点节中参考天线的物理位置及其当前功能。如果您的用例需要其他功能、额外的定位服务或更精确的天线位置，请<## aws-groundstation@amazon.com> 联系我们。

要使用其中一 AWS Ground Station 根天线，您必须在特定位置预留时间。此预订被称为联系人。要成功安排联系， AWS Ground Station 需要其他数据才能确保联系成功。

- 您的卫星必须登载到一个或多个地点 — 这样可以确保您获得在请求的地点操作各种功能的许可。
- 您的卫星必须具有有效的星历——这样可以确保天线具有视线，并且在接触过程中可以准确地指向您的卫星。
- 您必须拥有有效的任务配置文件 — 这允许您自定义此联系人的行为，包括如何接收和向卫星发送数据。您可以为同一辆车使用多个任务配置文件来创建不同的触点，以适应您遇到的不同操作姿势或场景。

卫星上线

将卫星载入 AWS Ground Station 是一个多步骤的过程，包括数据收集、技术验证、频谱许可，以及集成和测试。本指南的 “卫星入门” 部分将引导您完成此过程。

任务概况构成

卫星频率信息、数据平面信息和其他详细信息封装在任务配置文件中。任务配置文件是配置组件的集合。这使您可以根据自己的用例在不同的任务配置文件中重复使用配置组件。由于任务概况不直接引用单个卫星，而只包含有关其技术能力的信息，因此具有相同配置的多颗卫星也可以重复使用任务配置文件。

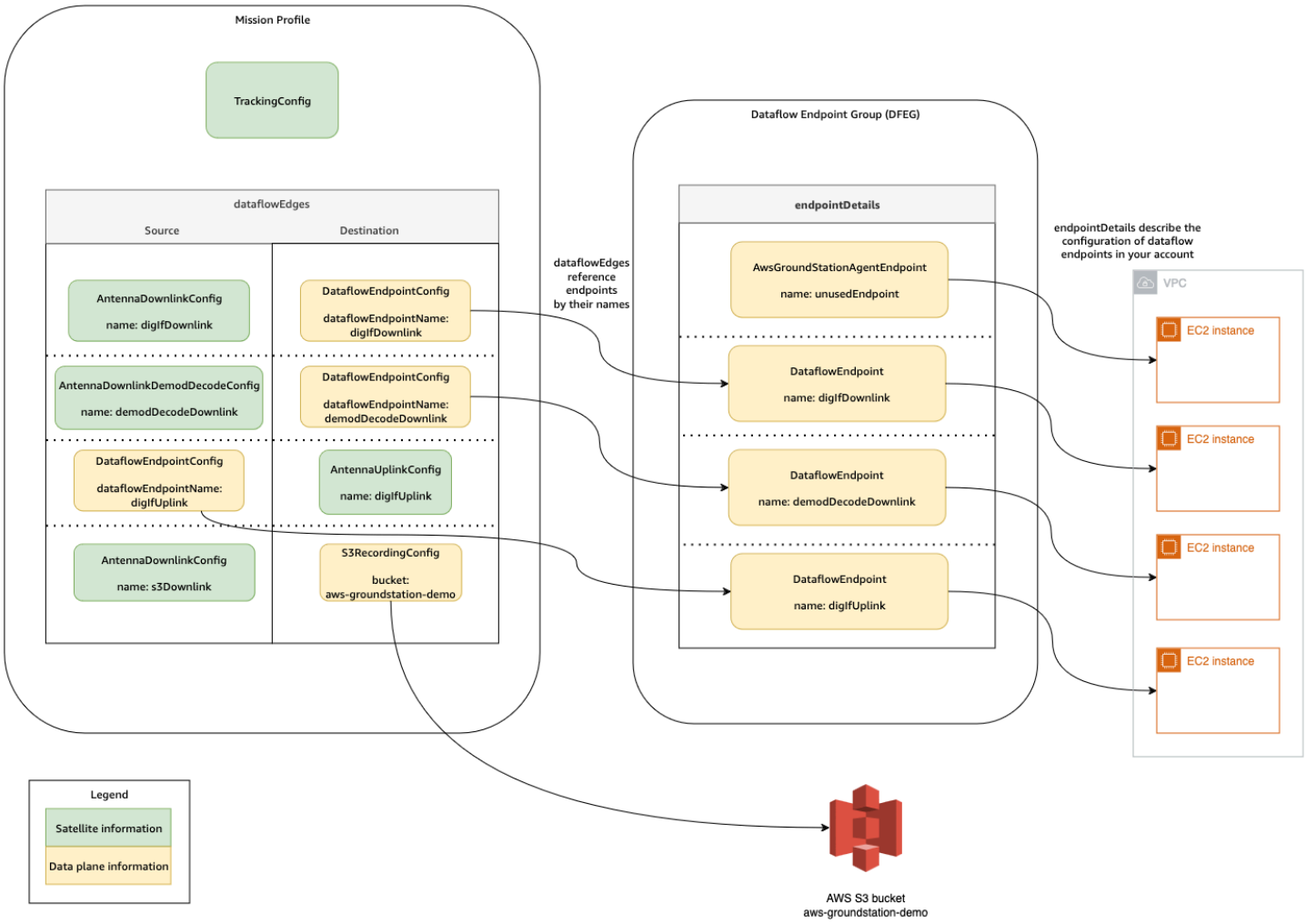
有效的任务配置文件将包含跟踪配置和一个或多个数据流。跟踪配置将指定您在联系期间的跟踪偏好。数据流中的每个配置对都会建立源和目标。根据您的卫星及其运行模式，任务配置文件中数据流的确切数量将有所不同，以代表您的上行和下行链路通信路径以及任何数据处理方面。

- 有关配置将在联系期间使用的 Amazon VPC、Amazon S3 和亚马逊 EC2 资源的更多信息，请参阅使用数据流。
- 有关每个配置的行为方式的详细信息，请参阅使用 AWS Ground Station 配置。

- 有关所有预期参数的具体详细信息，请参阅使用 AWS Ground Station 任务档案。
- 有关如何创建各种任务配置文件以支持您的用例的示例，请参阅任务配置文件配置示例。

下图显示了任务概况示例和所需的额外资源。请注意，该示例显示了一个名为 unuseDendPoint 的任务配置文件不需要的数据流端点，以证明其灵活性。该示例支持以下数据流：

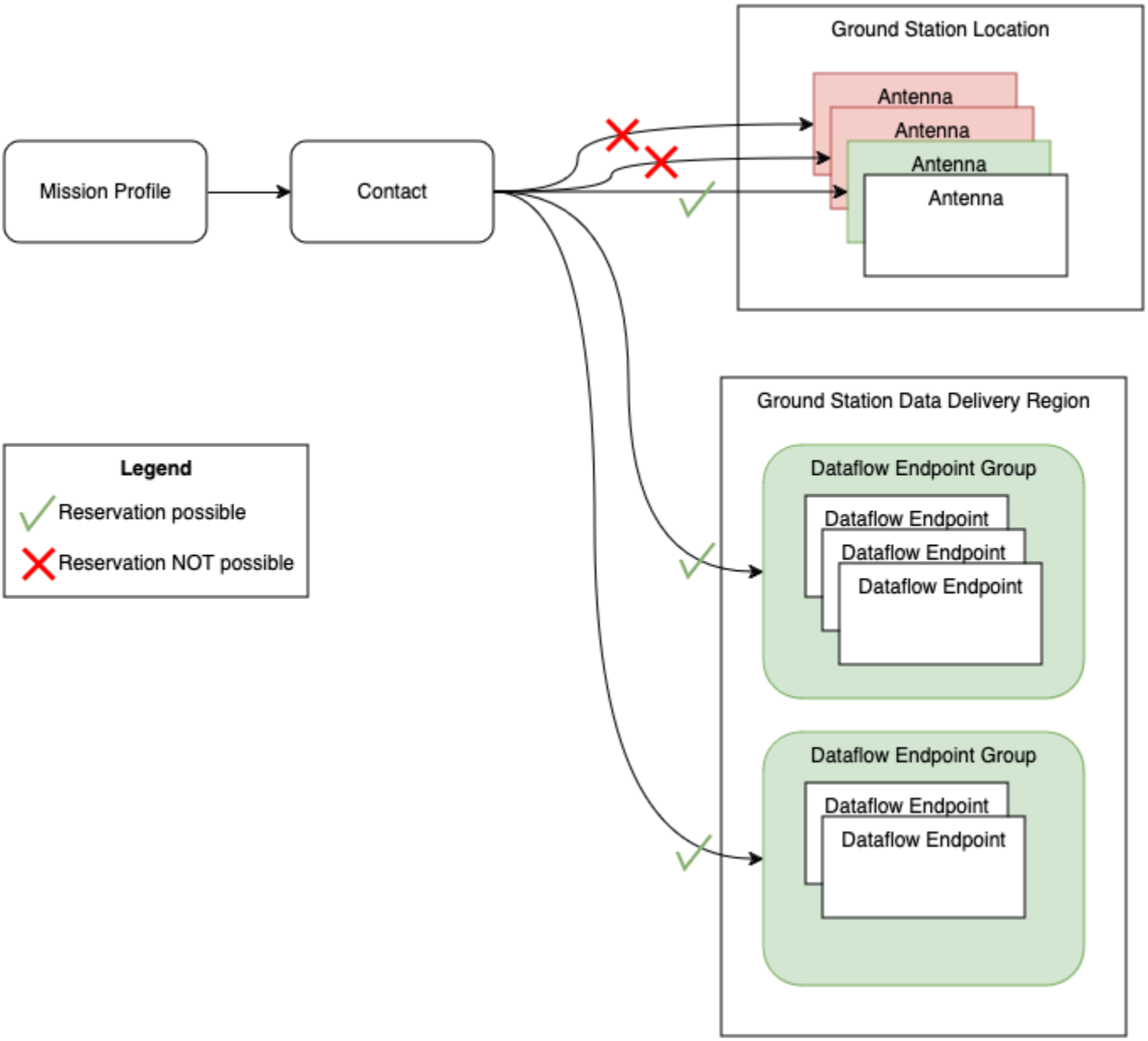
- 将数字中频数据同步下行链路传输到您管理的 Amazon EC2 实例。用名字digIfDownlink表示。
- 将数字中频数据异步下行链路传输到 Amazon S3 存储桶。用存储桶名称aws-groundstation-demo表示。
- 将解调和解码后的数据同步下行链接到您管理的 Amazon EC2 实例。用名字demodDecodeDownlink表示。
- 将数据从您管理的 Amazon EC2 实例同步上行链接到 AWS Ground Station 托管天线。用名字digIfUplink表示。



联系人日程安排

有了有效的任务配置文件，您就可以请求联系您的机载卫星。联系人预约请求是异步的，以便全球天线服务有时间在所有相关 AWS 地区实现一致的时间表。在此过程中，将对所请求的地面站位置的各种天线进行评估，以确定它们是否可用并能够处理接触。在此过程中，还将评估您配置的数据流端点以确定其可用性。评估进行期间，联系人状态将处于 “排程” 状态。

此异步调度过程将在请求发出后的五分钟内完成，但通常在一分钟内完成。请在安排利用事件 AWS Ground Station 实现自动化时间查看基于事件的监控。



可以执行且有空闲的联系会导致已安排的联系人。通过预定接触，您执行联系所需的资源已在您的任务配置文件所定义的 AWS 区域中预留。无法执行或部件不可用的联系将导致 FAILED_TO _SCHEDULE 联系失败。有关调试排查计划失败的联系人故障的详细信息，请参阅。

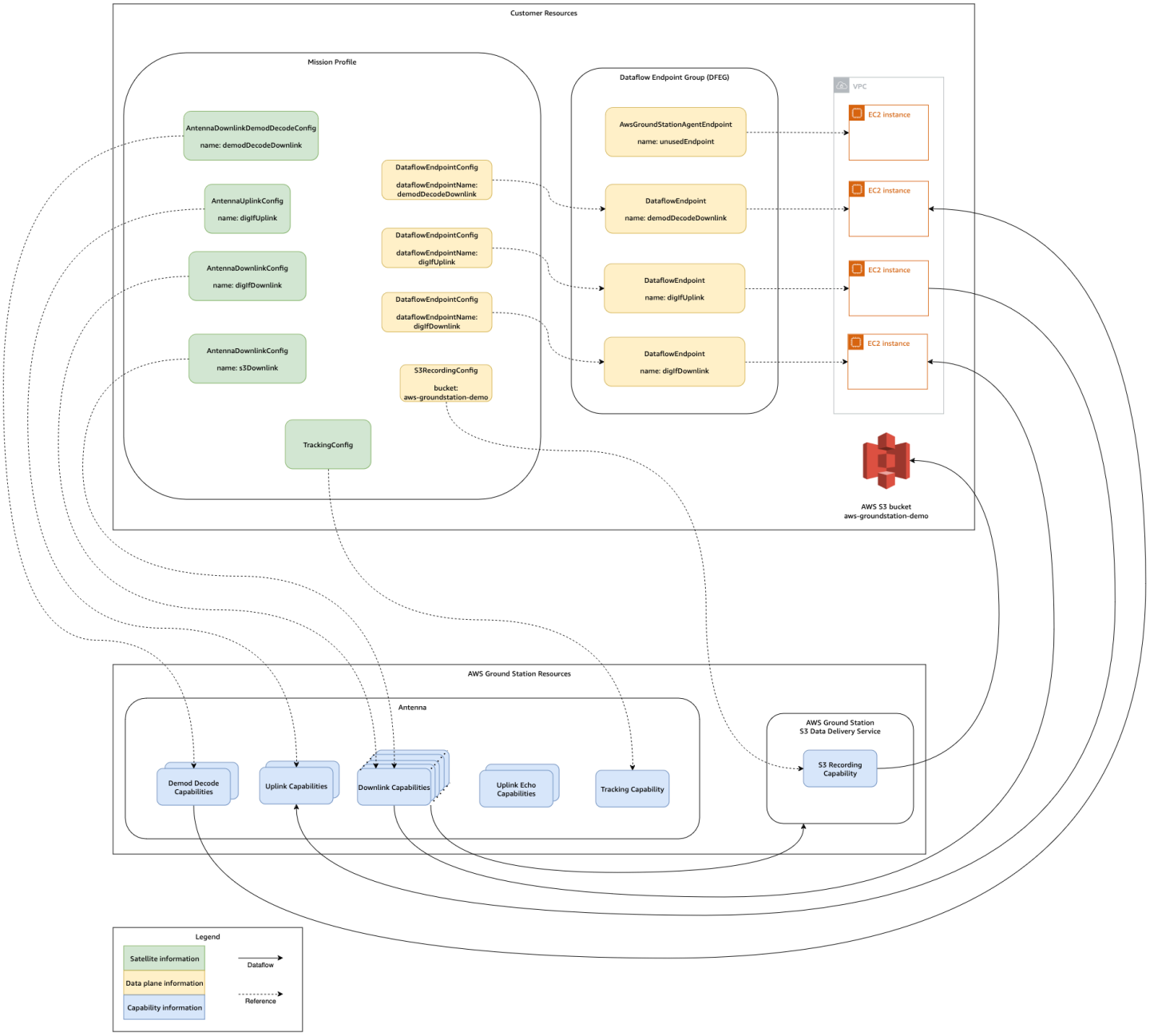
联系人执行

AWS Ground Station 将在您的联系预约期间自动编排您的 AWS 托管资源。如果适用，您负责协调任务配置文件中定义为数据流端点的 EC2 资源。 AWS Ground Station 提供 AWS Ev EventBridge en ts，用于自动编排资源以降低成本。有关更多信息，请参阅利用事件 AWS Ground Station 实现自动化。

在联系期间，我们会向 AWS CloudWatch 提供有关您的联系业绩的遥测数据。有关如何在执行期间监控您的联系人的信息，请参阅通过以下方式了解监控 AWS Ground Station。

下图延续了前面的示例，显示了在联系期间精心策划的相同资源。

> **Note**
>
> 本示例中并未使用所有天线功能。例如，每根天线都有十几种天线下行链路功能可用，支持多种频率和极化。有关 AWS Ground Station 天线提供的每种功能类型的数量及其支持的频率和极化的更多详细信息，请参阅。AWS Ground Station 网站能力



在联系结束时，AWS Ground Station 将评估您的联系表现，并确定最终的联系状态。未检测到错误的联系人将变为“已完成”联系状态。在联系期间服务错误导致数据传输问题的联系人将进入AWS_FAILED状态。在联系过程中，如果客户或用户错误导致数据传输问题，则联系人将进入失败状态。在裁决期间，不考虑接触时间以外的错误，即通过前或通过后的错误。

请参阅了解联系人生命周期了解更多信息。

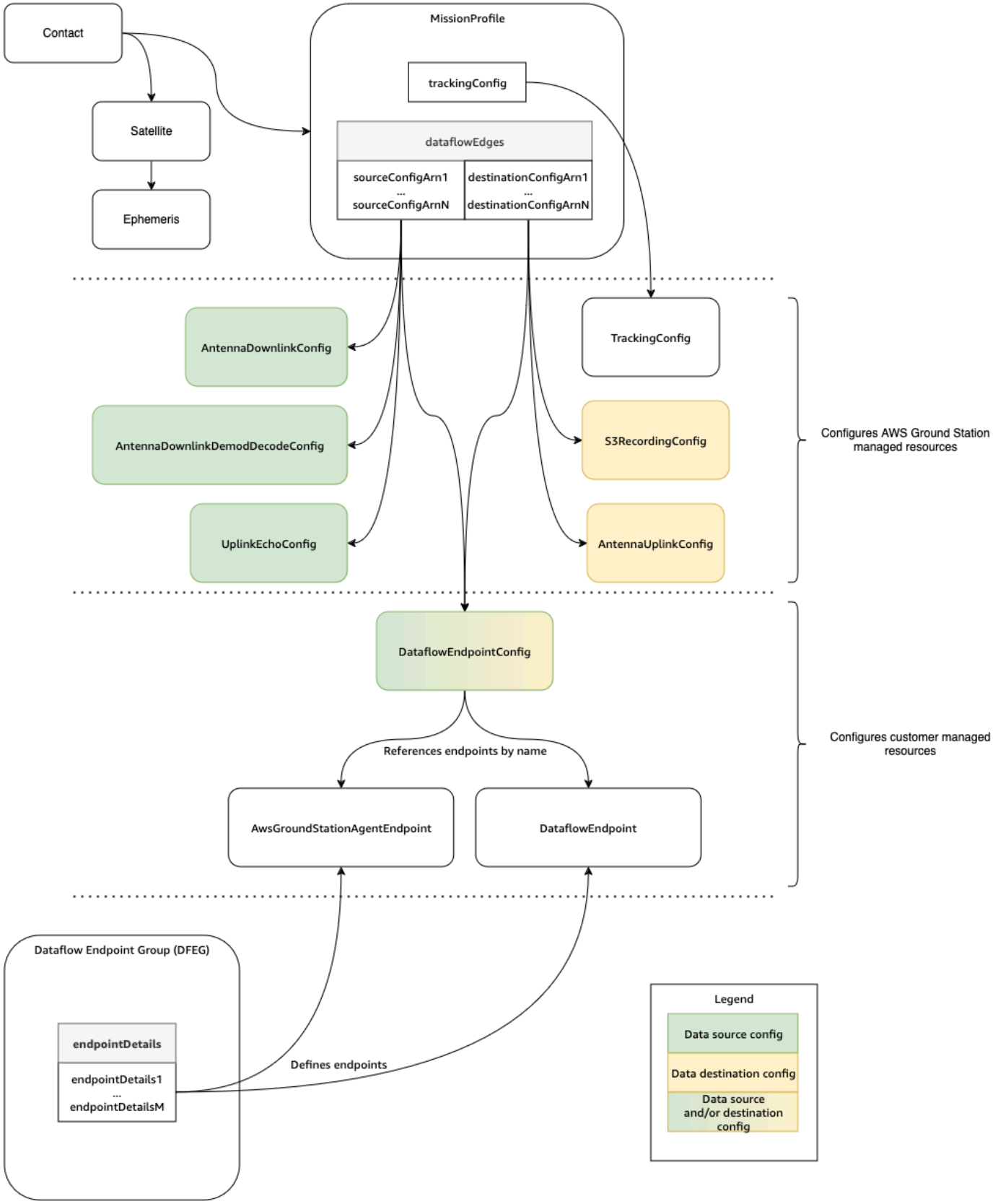
数字双胞胎

的数字双胞胎功能 AWS Ground Station 允许您根据虚拟地面站位置安排接触。这些虚拟地面站是生产地面站的精确复制品，包括天线功能、场地掩码和实际的 GPS 坐标。与生产地面站相比，数字双胞胎功能使您能够以低廉的成本测试联系人编排工作流程。参阅 使用 AWS Ground Station 数字双胞胎功能 了解更多信息。

了解 AWS Ground Station 核心组件

本节提供了 AWS Ground Station 核心组件的详细定义。

下图显示了的核心组件 AWS Ground Station 以及它们之间的关系。箭头表示组件之间依赖关系的方向，其中每个组件都指向其依赖关系。


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

以下主题详细描述了 AWS Ground Station 核心组件。

主题

使用 AWS Ground Station 任务档案

任务配置文件包含的配置和参数说明了如何执行联络。预留联络或搜索可用联络时，您提供打算使用的任务配置文件。任务配置文件将您的所有配置集合在一起，并定义了在联络期间数据将去往何处。

任务概况可以在具有相同无线电特性的卫星之间共享。您可以创建其他 Dataflow 端点组，以限制您要为星座执行的最大同时接触次数。

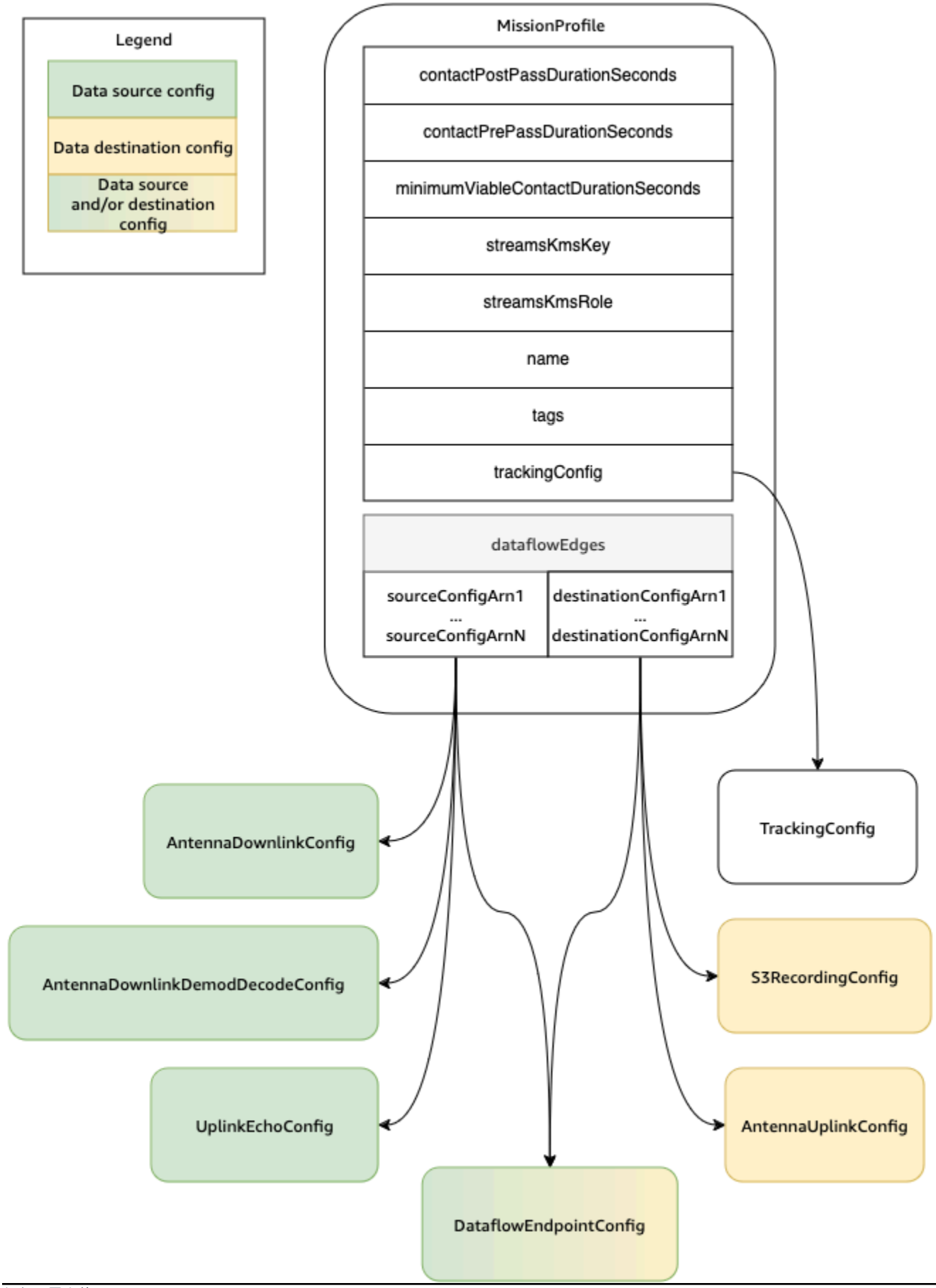
跟踪配置被指定为任务配置文件中的一个唯一字段。跟踪配置用于指定您在联系期间使用节目跟踪和自动跟踪的偏好。有关更多信息，请参阅 跟踪配置。

所有其他配置都包含在任务配置文件`dataflowEdges`字段中。这些配置可以看作是数据流节点，每个节点都代表可以发送或接收数据的 AWS Ground Station 托管资源及其相关配置。该`dataflowEdges`字段定义了需要哪些源和目标数据流节点（配置）。单个数据流边缘是两个配置 Amazon 资源名称 (ARNs) 的列表，第一个是源配置，第二个是目标配置。通过在两个配置之间指定数据流边缘，您可以分辨出联系期间数据应 AWS Ground Station 从何处流向何处。有关更多信息，请参阅 使用 AWS Ground Station 配置。

`contactPrePassDurationSeconds`和`contactPostPassDurationSeconds`允许您指定与联系人相关的接收 CloudWatch 事件通知的时间。有关与您的联系人相关的事件的时间表，请阅读了解联系人生命周期。

任务配置文件的 `name` 字段有助于区分您创建的任务配置文件。

`streamsKmsRole`和`streamsKmsKey`用于定义通过 A AWS Ground Station gent 传输数据时使用的 AWS Ground Station 加密。请参阅 传输期间的数据加密 AWS Ground Station。


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

参数和示例的完整列表包含在以下文档中。

- AWS::GroundStation::MissionProfile CloudFormation 资源类型

使用 AWS Ground Station 配置

配置是用于定义联系人各个方面的参数的资源。 AWS Ground Station 将您想要的配置添加到任务配置文件中，然后在执行联络时使用该任务配置文件。您可以定义几种不同类型的配置。这些配置可以分为两类：

- 跟踪配置
- 数据流配置

A TrackingConfig是唯一的跟踪配置类型。它用于在接触期间配置天线的自动跟踪设置，并且在任务配置文件中是必需的。

可以在任务配置文件数据流中使用的配置可以看作是数据流节点，每个节点都代表可以发送或接收数据的 AWS Ground Station 托管资源。任务配置文件至少需要一对这样的配置，其中一个代表数据源，一个代表目的地。下表汇总了这些配置。

Config 名称	数据流源/目的地
AntennaDownlinkConfig	来源
AntennaDownlinkDemodDecodeConfig	来源
UplinkEchoConfig	来源
S3 RecordingConfig	目标
AntennaUplinkConfig	目标
DataflowEndpointConfig	来源和/或目的地

有关如何使用 AWS CloudFormation、或 AWS Ground Station API 对配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface下面还提供了针对特定配置类型文档的链接。

- AWS::GroundStation::Config CloudFormation 资源类型

- Config AWS CLI 参考
- 配置 API 参考

跟踪配置

您可以使用任务配置文件中的跟踪配置来确定是否应在您的联络期间启用自动跟踪。此配置只有一个参数：autotrack。autotrack 参数可能具有以下值：

- REQUIRED：您的联络需要自动跟踪。
- PREFERRED：最好对联络启用自动跟踪，但如果不使用，仍然可以执行联络。
- REMOVED：不应对您的联络启用自动跟踪。

AWS Ground Station 将使用编程跟踪，当不使用自动跟踪时，它会根据你的星历进行指向。有关星历构造方法了解如何 AWS Ground Station 使用卫星星历数据的详细信息，请参考。

在找到预期信号之前，Autotrack 将使用节目跟踪。一旦发生这种情况，它将继续根据信号的强度进行跟踪。

有关如何使用 AWS CloudFormation、或 AWS Ground Station API 对跟踪配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface

- AWS::GroundStation::Config TrackingConfig CloudFormation 财产
- Config AWS CLI 参考（参见trackingConfig -> (structure)部分）
- TrackingConfig API 参考

天线下行传输配置

您可在联络期间使用天线下行链路配置，以配置需要下行链路传输的天线。它们包括一个光谱配置，其中指定了下行联络期间应使用的频率、带宽和极化。

此配置表示数据流中的源节点。它负责对射频数据进行数字化。从该节点流出的数据将遵循信号数据/IP 格式。有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

如果您的下行传输用例需要解调或解码，请参阅 天线下行传输解调解码配置 。

有关如何使用 AWS CloudFormation、或 API 对天线下行链路配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::Config AntennaDownlinkConfig CloudFormation 财产
- Config AWS CLI 参考（参见antennaDownlinkConfig -> (structure)部分）
- AntennaDownlinkConfig API 参考

天线下行传输解调解码配置

天线下行链路演示解码配置是一种更复杂且可自定义的配置类型，可用于通过解调和/或解码来执行下行链路接触。如果您有兴趣执行这些类型的联系，请发送电子邮件<# aws-groundstation@amazon.com> 与 AWS Ground Station 团队联系。我们将帮助您确定适合您的用例的正确配置和任务配置文件。

此配置表示数据流中的源节点。它负责对射频数据进行数字化处理，并按照规定执行解调和解码。从该节点流出的数据将遵循Demodulated/Decoded Data/IP格式。有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

有关如何使用、或 API 对天线下行链路演示解码配置执行操作的更多信息 AWS CloudFormation，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::Config AntennaDownlinkDemodDecodeConfig CloudFormation 财产
- Config AWS CLI 参考（参见antennaDownlinkDemodDecodeConfig -> (structure)部分）
- AntennaDownlinkDemodDecodeConfig API 参考

天线上行传输配置

您可以在联络期间使用天线上行链路配置，以配置需要上行链路传输的天线。它们由包含频率、极化和目标有效各向同性辐射功率 (EIRP) 的频谱配置组成。有关如何配置上行回波的信息，请参阅 天线上行传输回波配置。

此配置表示数据流中的目标节点。它会将提供的数字化射频数据信号转换为模拟信号，然后将其发射给您的卫星接收。流向该节点的数据预计将符合信号数据/IP 格式。有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

有关如何使用 AWS CloudFormation、或 API 对天线上行链路配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::Config AntennaUplinkConfig CloudFormation 财产
- Config AWS CLI 参考（参见antennaUplinkConfig -> (structure)部分）

- AntennaUplinkConfig API 参考

天线上行传输回波配置

上行传输回波配置告诉天线如何执行上行传输回波。上行链路回声可用于验证发送到航天器的命令，并执行其他高级任务。这是通过记录 AWS Ground Station 天线（即上行链路）传输的实际信号来实现的。这会回声天线发送回您的数据流端点的信号，并且应该与传输的信号相匹配。上行传输回波配置包含上行传输配置的 ARN。天线使用在执行上行传输回波时 ARN 指向的上行传输配置中的参数。

此配置表示数据流中的源节点。从该节点流出的数据将符合信号数据/IP 格式。有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

有关如何使用 AWS CloudFormation、或 API 对上行链路 echo 配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::Config UplinkEchoConfig CloudFormation 财产
- Config AWS CLI 参考（参见`uplinkEchoConfig -> (structure)`部分）
- UplinkEchoConfig API 参考

数据流端点配置

> **Note**
>
> Dataflow 终端节点配置仅用于向亚马逊 EC2 传输数据，不用于向 Amazon S3 传输数据。

您可以使用数据流端点配置来指定联络期间从数据流端点组中的哪个数据流端点流入或流向哪个数据流端点。数据流端点配置的两个参数指定数据流端点的名称和区域。预订联系人时， AWS Ground Station 会分析您指定的任务配置文件并尝试在 AWS 区域内找到一个数据流端点组，该组包含任务配置文件中包含的数据流端点配置所指定的所有数据流端点。如果找到了合适的数据流端点组，则联系状态将变为 “已计划”，否则将变为 FAILED_TO_SCHEDULE。有关联系人可能的状态的更多信息，请参阅AWS Ground Station 联系人状态。

数据流端点配置的 `dataflowEndpointName` 属性指定联络期间从数据流端点组中的哪个数据流端点流入或流向哪个数据流端点。

`dataflowEndpointRegion` 属性指定数据流端点所在的区域。如果在您的数据流终端节点配置中指定了区域，则会在指定区域中 AWS Ground Station 查找数据流终端节点。如果未指定区域， AWS

Ground Station 则默认为联系人的地面站区域。如果您的数据流端点的区域与联络的地面站区域不同，则该联络被视为跨区域数据传输联络。使用数据流有关跨区域数据流的更多信息，请参阅。

有关数据流使用 AWS Ground Station 数据流终端节点组的不同命名方案如何使您的用例受益的提示，请参阅。

有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

有关如何使用、或 API 对数据流端点配置执行操作的更多信息 AWS CloudFormation，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::Config DataflowEndpointConfig CloudFormation 财产
- Config AWS CLI 参考（参见`dataflowEndpointConfig -> (structure)`部分）
- DataflowEndpointConfig API 参考

亚马逊 S3 录音 Config

> **Note**
>
> Amazon S3 记录配置仅用于向亚马逊 S3 传输数据，不用于向亚马逊 EC2传输数据。

此配置表示数据流中的目标节点。该节点会将来自数据流源节点的传入数据封装到 pcap 数据中。有关如何使用此配置构造数据流的更多详细信息，请参阅 使用数据流

您可以使用 S3 记录配置来指定要将下行链接数据以及使用的命名约定传送到的 Amazon S3 存储桶。以下内容指定了有关这些参数的限制和详细信息：

- Amazon S3 存储桶的名称必须以 `aws-groundstation` 开始。
- IAM 角色必须拥有信任策略，以允许 `groundstation.amazonaws.com` 服务主体承担该角色。有关示例，请参阅下面的示例信任策略部分。在配置创建过程中，配置资源 ID 不存在，信任策略必须使用星号 (*) 代替，*`your-config-id`*并且可以在创建后使用配置资源 ID 进行更新。

示例信任策略

有关如何更新角色的信任策略的更多信息，请参阅 IAM 用户指南中的管理 IAM 角色。

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

- IAM 角色必须拥有 IAM policy，以允许该角色对存储桶执行 `s3:GetBucketLocation` 操作以及对存储桶对象执行 `s3:PutObject` 操作。如果 Amazon S3 存储桶拥有存储桶策略，则该存储桶策略还必须允许 IAM 角色执行这些操作。有关示例，请参阅下面的示例角色策略部分。

示例角色策略

有关如何更新或附加角色策略的更多信息，请参阅 IAM 用户指南中的管理 IAM 策略。

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

- 命名 S3 数据对象时将使用前缀。您可以指定可选的密钥进行替换，这些值将替换为联系人详细信息中的相应信息。例如，前缀`{satellite_id}/{year}/{month}/{day}`将被替换，其结果将是类似的输出 `fake_satellite_id/2021/01/10`

  替换的可选密钥：`{satellite_id}{config-name}| {config-id} | {year} | {month} | {day} |`

有关如何使用 AWS CloudFormation、或 AWS Ground Station API 对 S3 录制配置执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface

- AWS::GroundStation::Config S3 RecordingConfig CloudFormation 属性
- Config AWS CLI 参考（参见`s3RecordingConfig -> (structure)`部分）
- S3 RecordingConfig API 参考

使用 AWS Ground Station 数据流终端节点组

Dataflow 端点定义了您希望在联系期间同步流入或流出数据的位置。数据流终端节点总是作为数据流终端节点组的一部分创建的。在组中包括多个数据流终端节点，即表示您主张那些指定的终端节点都可以在单次联络中一起使用。例如，如果联络需要向三个独立的数据流端点发送数据，则单个数据流端点组中必须有三个端点与您的任务配置文件中的数据流端点配置相匹配。

> **Tip**
>
> 执行联系人时，数据流端点由您选择的名称标识。这些名称在整个账户中不必是唯一的。这允许使用相同的任务配置文件在不同的卫星和天线上同时执行多个接触。如果您拥有一组具有相同运行特性的卫星，这可能会很有用。您可以向上扩展 Dataflow 端点组的数量，以适应您的卫星星座所需的最大同步联系数量。

当数据流终端节点组中的一个或多个资源正用于某个联络时，在该次联络期间整个组都予以保留。您可以同时执行多个联络，但这些联络必须在不同的数据流端点组中执行。

> **Important**
>
> 数据流端点组必须处于 HEALTHY 状态，才能用于安排联络。有关如何对未处于HEALTHY状态的数据流终端节点组进行故障排除的信息，请参阅。排除 DataflowEndpointGroups 未处于正常状态的故障

有关如何使用 AWS CloudFormation、或 API 对数据流终端节点组执行操作的更多信息，请参阅以下文档。 AWS Command Line Interface AWS Ground Station

- AWS::GroundStation::DataflowEndpointGroup CloudFormation 资源类型
- 数据流终端节点组参考 AWS CLI
- 数据流端点组 API 参考

数据流终端节点

数据流终端节点组的成员是数据流终端节点。数据流端点有两种类型：AWS Ground Station 代理端点和数据流端点。对于这两种类型的端点，您将在创建 dataflow 端点组之前创建支持结构（例如 IP 地址）。使用数据流有关使用哪种 dataflow 端点类型以及如何设置支持结构的建议，请参阅。

以下各节描述了两种支持的端点类型。

> **Important**
>
> 单个数据流终端节点组中的所有数据流端点必须属于同一类型。您不能在同一个组中将AWS Ground Station 代理端点与 Dataflow 端点混合使用。如果您的用例需要两种类型的终端节点，则必须为每种类型创建单独的 dataflow 端点组。

AWS Ground Station 代理端点

AWS Ground Station 代理端点使用 AWS Ground Station 代理作为软件组件来终止连接。当您想要下行链路超过 50% MHz 的数字信号数据时，请使用 AWS Ground Station 代理数据流端点。要构造 AWS Ground Station 代理终端节点，您只需要填充的`AwsGroundStationAgentEndpoint` EndpointDetails字段。有关 AWS Ground Station 代理的更多信息，请参阅完整的《AWS Ground Station 代理用户指南》。

`AwsGroundStationAgentEndpoint` 由以下内容组成：

- `Name`-数据流端点名称。要使联系人使用此数据流端点，此名称必须与您的数据流端点配置中使用的名称相匹配。
- `EgressAddress`-用于从代理输出数据的 IP 和端口地址。
- `IngressAddress`-用于将数据传入代理的 IP 和端口地址。

数据流端点

Dataflow Endpoint 使用网络应用程序作为软件组件来终止连接。当您想要上行链路数字信号数据、下行链路少于 50% MHz 的数字信号数据或下行链路解调/解码的信号数据时，请使用 Dataflow Endpoint。要构建 Dataflow Endpoint，您需要填充`Endpoint`和`Security Details`字段。EndpointDetails

`Endpoint` 由以下内容组成：

- `Name`-数据流端点名称。要使联系人使用此数据流端点，此名称必须与您的数据流端点配置中使用的名称相匹配。
- `Address`-使用的 IP 和端口地址。

`SecurityDetails` 由以下内容组成：

- `roleArn`- AWS Ground Station 将担任在您的 VPC 中创建弹性网络接口 () 的角色的亚马逊资源名称 (ARNENIs)。它们 ENIs 充当联系期间流式传输的数据的入口和输出点。
- `securityGroupIds`：附加到弹性网络接口的安全组。
- `subnetIds`- AWS Ground Station 可能放置弹性网络接口以向您的实例发送流的子网列表。如果指定了多个子网，则这些子网必须可以相互路由。如果子网位于不同的可用区 (AZs)，则可能会产生跨可用区数据传输费用。

传递到 `roleArn` 的 IAM 角色必须拥有信任策略，以允许 `groundstation.amazonaws.com` 服务主体代入该角色。有关示例，请参阅下面的示例信任策略部分。在创建端点期间，端点资源 ID 不存在，因此信任策略必须使用星号 (*) 代替。*your-endpoint-id*这可以在创建后进行更新以使用端点资源 ID，从而将信任策略的范围限定为该特定的数据流端点组。

IAM 角色必须具有 AWS Ground Station 允许设置的 IAM 策略 ENIs。有关示例，请参阅下面的示例角色策略部分。

示例信任策略

有关如何更新角色的信任策略的更多信息，请参阅 IAM 用户指南中的管理 IAM 角色。

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

```
}
```

示例角色策略

有关如何更新或附加角色策略的更多信息，请参阅 IAM 用户指南中的管理 IAM 策略。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
        "ec2:CreateNetworkInterfacePermission",
        "ec2:DeleteNetworkInterfacePermission",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs",
        "ec2:DescribeSecurityGroups"
      ]
    }
  ]
}
```

使用 AWS Ground Station 代理

AWS Ground Station 代理使您能够在 AWS Ground Station 联系期间接收（下行链路）同步宽带数字中频 (digiF) 数据流。

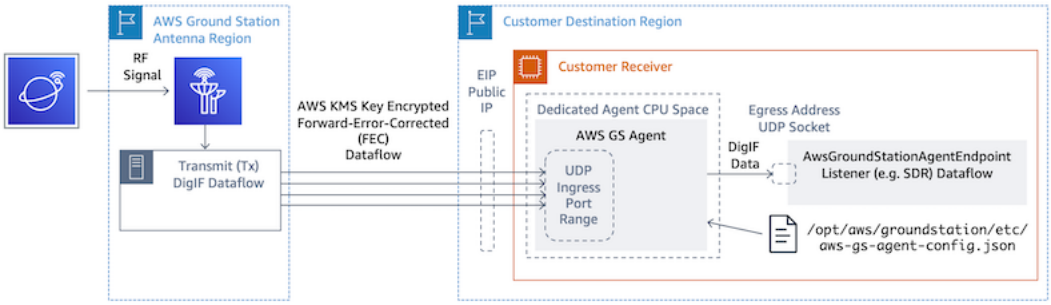
工作方式

您可以选择两个数据传输选项：

1. 向 EC2 实例传输数据-将数据传输到您拥有的 EC2 实例。由您管理代 AWS Ground Station 理。如果您需要近乎实时的数据处理，则选择此选项。有关 EC2 数据传输的信息，请参阅使用数据流部分。
2. 向 S3 存储桶传输数据-向 AWS S3 存储桶传输的数据完全由管理 AWS Ground Station。有关 S3 数据传输的信息，请参阅 开始使用 指南。

两种数据传输模式都需要您创建一组 AWS 资源。强烈建议使用 CloudFormation 来创建 AWS 资源，以确保可靠性、准确性和可支持性。每个联系人只能向 EC2 或 S3 传输数据，但不能同时向两者传送数据。

下图显示了使用软件定义无线电 (SDR) 或类似监听器从 AWS Ground Station 天线区域到您的 EC2 实例的 digiF 数据流。



其他信息

有关更多详细信息，请参阅完整的《AWS Ground Station 代理用户指南》。

开始使用

在开始之前，您应该熟悉中的基本概念。 AWS Ground Station有关更多信息，请参阅 如何 AWS Ground Station 运作。

以下是 AWS Identity and Access Management (IAM) 的最佳实践以及您需要的权限。设置适当的角色后，您可以开始执行其余步骤。

注册获取 AWS 账户

如果您没有 AWS 账户，请完成以下步骤来创建一个。

要注册 AWS 账户

1. 打开https://portal.aws.amazon.com/billing/注册。
2. 按照屏幕上的说明操作。

   在注册时，将接到电话或收到短信，要求使用电话键盘输入一个验证码。

   当您注册时 AWS 账户，就会创建AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务 和资源。作为最佳安全实践，请为用户分配管理访问权限，并且只使用根用户来执行需要根用户访问权限的任务。

AWS 注册过程完成后会向您发送一封确认电子邮件。您可以随时前往 https://aws.amazon.com/并选择“我的账户”，查看您当前的账户活动并管理您的账户。

创建具有管理访问权限的用户

注册后，请保护您的安全 AWS 账户 AWS 账户根用户 AWS IAM Identity Center，启用并创建管理用户，这样您就不会使用 root 用户执行日常任务。

保护你的 AWS 账户根用户

1. 选择 Root 用户并输入您的 AWS 账户 电子邮件地址，以账户所有者的身份登录。AWS Management Console在下一页上，输入您的密码。

   要获取使用根用户登录方面的帮助，请参阅《AWS 登录 用户指南》中的 Signing in as the root user。

2. 为您的根用户启用多重身份验证（MFA）。

   有关说明，请参阅 I A M 用户指南中的为 AWS 账户 根用户启用虚拟 MFA 设备（控制台）。

创建具有管理访问权限的用户

1. 启用 IAM Identity Center。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Enabling AWS IAM Identity Center。

2. 在 IAM Identity Center 中，为用户授予管理访问权限。

   有关使用 IAM Identity Center 目录 作为身份源的教程，请参阅《用户指南》 IAM Identity Center 目录中的使用默认设置配置AWS IAM Identity Center 用户访问权限。

以具有管理访问权限的用户身份登录

- 要使用您的 IAM Identity Center 用户身份登录，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录网址。

  有关使用 IAM Identity Center 用户登录的帮助，请参阅AWS 登录 用户指南中的登录 AWS 访问门户。

将访问权限分配给其他用户

1. 在 IAM Identity Center 中，创建一个权限集，该权限集遵循应用最低权限的最佳做法。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Create a permission set。

2. 将用户分配到一个组，然后为该组分配单点登录访问权限。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Add groups。

为您的 AWS 账户添加 AWS Ground Station 权限

要在不需要管理员用户 AWS Ground Station 的情况下使用，您需要创建一个新策略并将其附加到您的 AWS 账户。

1. 登录 AWS Management Console 并打开 IAM 控制台。

2. 创建新策略。使用以下步骤：

   a. 在导航窗格中选择策略，然后选择创建策略。

   b. 在 JSON 选项卡中，使用以下值之一编辑 JSON。使用最适合您的应用程序的 JSON。

   - 对于 Ground Station 管理权限，将操作设置为 groundstation:*，如下所示：

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

   - 对于只读权限，请将操作设置为 groundstation:Get*、groundstation:List* 和 groundstation:Describe*，如下所示：

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

- 为了通过多因素身份验证提高安全性，请将 “操作” 设置为 g roundstation: *，将 Condition / Bool 设置为 aws:: true，如下所示：MultiFactorAuthPresent

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

3. 在 IAM 控制台中，将您创建的策略附加到所需用户。

有关 IAM 用户并附加策略的更多信息，请参阅 IAM 用户指南。

机载卫星

将卫星载入 AWS Ground Station 是一个多步骤的过程，包括数据收集、技术验证、频谱许可，以及集成和测试。还需要签订保密协议 (NDAs)。

客户入职流程概述

卫星载入是一个手动流程，可以在 AWS Ground Station 控制台页面的 “卫星和资源” 部分找到。以下描述了整个过程。

1. 查看本AWS Ground Station 地点节以确定您的卫星是否符合地理和无线电频率特征。
2. 要开始将您的卫星载入 AWS Ground Station，请发送电子邮件<# aws-groundstation@amazon.com>，简要概述您的任务和卫星需求，包括您的组织名称、所需的频率、卫星的发射时间或发射时间、卫星的轨道类型以及您是否计划使用使用 AWS Ground Station 数字双胞胎功能。

3. 您的申请经过审核和批准后， AWS Ground Station 将在您计划使用的特定地点申请监管许可。此步骤的持续时间将因地点和任何现行法规而异。
4. 获得批准后，您的卫星将可见供您使用。 AWS Ground Station 将向您发送更新成功通知。

（可选）命名卫星

入职后，您可能需要在卫星记录中添加一个名称，以便更轻松地识别它。使用 “联系人” 页面时， AWS Ground Station 控制台能够显示用户定义的卫星名称以及 Norad ID。显示卫星名称可以在计划时更容易选择正确的卫星。为此，可以使用标签。

标记 AWS Ground Station 卫星可以通过标签资源 API 和 AWS CLI 或其中一个 AWS 来完成。 SDKs 本指南将介绍如何使用 AWS Ground Station CLI 标记公共广播卫星 Aqua（Norad ID 27424）。`us-west-2`

AWS Ground Station CLI

AWS CLI 可以用来与之交互 AWS Ground Station。在 AWS CLI 使用标记卫星之前，必须满足以下 AWS CLI 先决条件：

- 确保 AWS CLI 已安装。有关安装的信息 AWS CLI，请参阅安装 AWS CLI 版本 2。
- 确保 AWS CLI 已配置。有关配置的信息 AWS CLI，请参阅配置 AWS CLI 版本 2。
- 将您常用的配置设置和凭证保存在由 AWS CLI维护的文件中。您需要这些设置和凭据来保留和管理您的 AWS Ground Station 联系人 AWS CLI。有关保存配置和凭据设置的更多信息，请参阅配置和凭据文件设置。

配置完成并可供使用后 AWS CLI ，请查看 AWS Ground Station CLI 命令参考页面，熟悉可用的命令。使用此服务时，请遵循 AWS CLI 命令结构，并在命令前加上前缀`groundstation` AWS Ground Station 以指定要使用的服务。有关 AWS CLI 命令结构的更多信息，请参阅 AWS CLI 页面中的命令结构。下面提供了一个示例命令结构。

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

命名卫星

首先，您需要获取要标记的卫星的 ARN。这可以通过 AWS C LI 中的列表卫星 API 来完成：

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

运行上述 CLI 命令将返回类似于以下内容的输出：

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

找到您要标记的卫星，并记下 satelliteArn。标记的一个重要注意事项是，标签资源 API 需要区域性 ARN，而列表卫星返回的 ARN 是全球性的。在下一步中，您应该使用您希望看到标签的区域（可能是您安排的区域）来扩展 ARN。对于该示例，我们将使用 us-west-2。此次更改后，ARN 将从：

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

到：

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

为了在控制台中显示卫星名称，卫星必须有一个 “Name" 标签作为密钥。此外，由于我们使用的是 AWS CLI，因此必须使用反斜杠对引号进行转义。标签将类似于：

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

接下来，您将调用标签资源 API 来标记卫星。这可以用 AWS CLI 这样的方法来完成：

```
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"}'
```

完成此操作后，您将能够在 AWS Ground Station 控制台中看到为卫星设置的名称。

更改卫星的名称

如果要更改卫星的名称，则只需使用相同的“Name”密钥再次使用卫星 ARN 调用 tag-resou rce，但在标签中使用不同的值即可。这将更新现有标记并在控制台中显示新名称。调用示例如下：

```
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"}'
```

删除卫星的名称

可以使用 untag- resource API 删除为卫星设置的名称。此 API 需要带有标记所在区域的卫星 ARN 以及标记密钥列表。对于名称，标记密钥为 “Name”。使用 AWS CLI 调用该 API 的示例如下所示：

```
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
```

公共广播卫星

除了登陆自己的卫星外，您还可以申请使用支持的公共广播卫星登机，这些卫星提供可公开访问的下行链路通信路径。这使您 AWS Ground Station 能够使用下行链路来自这些卫星的数据。

> **Note**
>
> 您将无法通过上行链路连接到这些卫星。您将只能使用可公开访问的下行链路通信路径。

AWS Ground Station 支持载入以下卫星以下行链路直接广播数据：

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

这些卫星一旦登机，就可以立即使用了。 AWS Ground Station 维护了许多预配置的 AWS CloudFormation 模板，以便更轻松地开始使用该服务。任务配置文件配置示例有关 AWS Ground Station 如何使用的示例，请参阅。

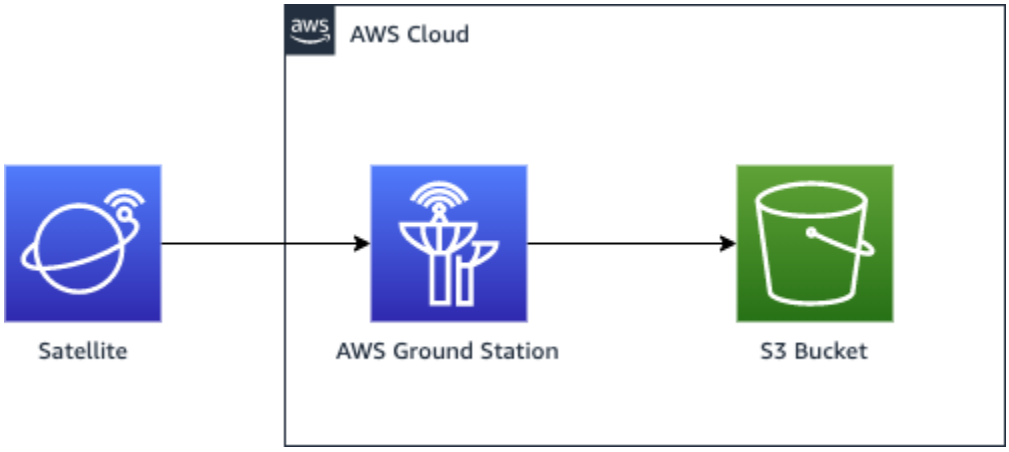
有关这些卫星及其传输的数据种类的更多信息，请参阅 Aqua、JPSS-1/NOAA-20 和 SNPP 以及 Terra。

规划您的数据流通信路径

您可以为卫星上的每条通信路径选择同步和异步通信。根据您的卫星和用例，您可能需要一种或两种类型。同步通信路径允许近乎实时的上行链路以及窄带和宽带下行链路操作。异步通信路径仅支持窄带和宽带下行链路操作。

异步数据传输

将数据传输至 Amazon S3 后，您的联络数据将异步传输到您账户中的 Amazon S3 存储桶。您的联络数据以数据包捕获 (pcap) 文件的形式传输，以允许将联络数据重放到软件定义无线电 (SDR) 中，或者从 pcap 文件中提取有效载荷数据进行处理。当天线硬件收到联络数据时，每 30 秒将 pcap 文件传输至您的 Amazon S3 存储桶，以便在联络期间需要时处理联络数据。收到数据后，您可以使用自己的后处理软件或使用其他 AWS 服务（例如 Amazon A SageMaker I 或 Amazon Rekognition）来处理数据。数据传输至 Amazon S3 仅适用于从您的卫星下载数据；无法将数据从 Amazon S3 上传到您的卫星。



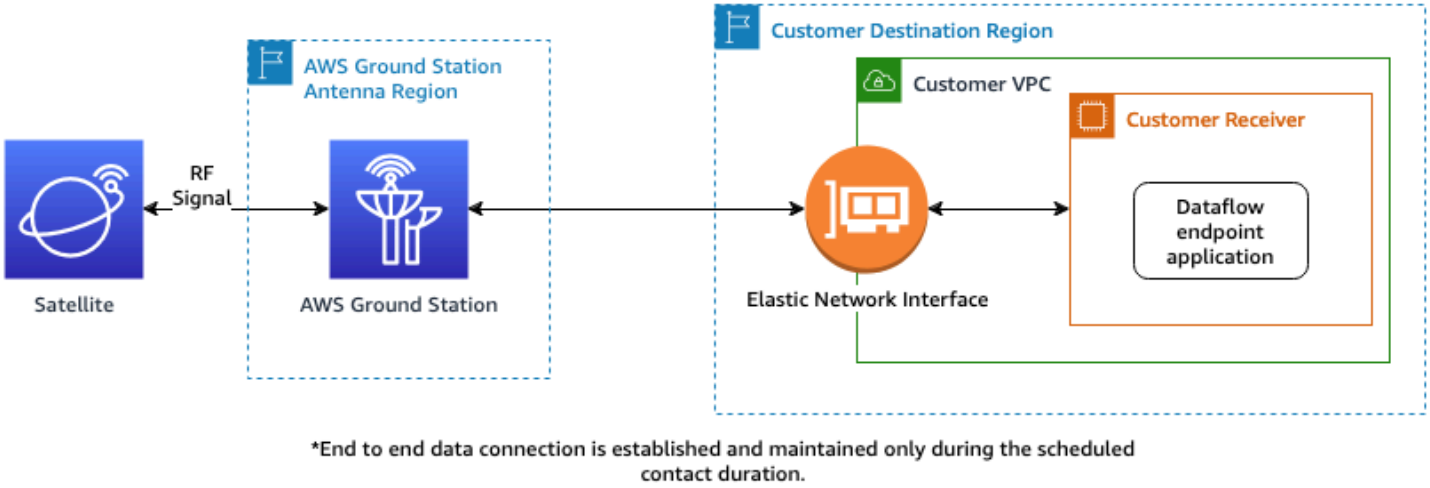
要使用此路径，您需要为创建一个 Amazon S3 存储桶 AWS Ground Station ，以便将数据传送到。在下一步中，您还需要在下一步中创建 S3 Recording Config。有关存储桶命名的限制以及如何指定文件使用的命名约定，请参阅。亚马逊 S3 录音 Config

同步数据传输

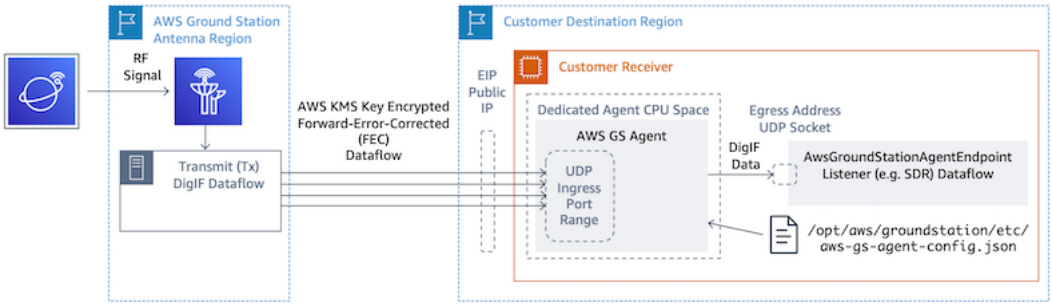
通过将数据传输到亚马逊 EC2，您的联系人数据将流入和流出您的亚马逊 EC2 实例。您可以在 Amazon EC2 实例上实时处理数据，也可以转发数据进行后期处理。

要使用同步路径，您需要设置和配置您的 Amazon EC2 实例，并创建一个或多个 Dataflow 终端节点组。要配置您的 Amazon EC2 实例，请参阅设置和配置 Amazon EC2。要创建您的 Dataflow 端点组，请参考。使用 AWS Ground Station 数据流终端节点组

如果您使用的是数据流端点配置，则下面显示了通信路径。



以下显示了使用 AWS Ground Station 代理配置时的通信路径。



创建配置

通过此步骤，您已经根据需要确定了卫星、通信路径以及 IAM、Amazon EC2 和 Amazon S3 资源。在此步骤中，您将创建存储各自参数的 AWS Ground Station 配置。

数据传输配置

要创建的第一个配置与您希望数据交付的地点和方式有关。使用上一步中的信息，您将构造以下许多配置类型。

- 亚马逊 S3 录音 Config-将数据传输到您的 Amazon S3 存储桶。
- 数据流端点配置-将数据传输到您的 Amazon EC2 实例。

卫星配置

卫星配置与 AWS Ground Station 如何与您的卫星通信有关。您将引用您在其中收集的信息机载卫星。

- 跟踪配置-设置在接触期间如何对车辆进行物理跟踪的偏好。这是构建任务概况所必需的。
- 天线下行传输配置-提供数字化的射频数据。
- 天线下行传输解调解码配置 -提供解调和解码后的射频数据。
- 天线上行传输配置-将数据上行链接到您的卫星。
- 天线上行传输回波配置-传送上行链路信号数据的回声。

创建任务档案

通过上一步中构建的配置，您已经确定了如何跟踪卫星以及与卫星通信的可能方式。在此步骤中，您将构建一个或多个任务配置文件。任务配置文件表示将可能的配置汇总为预期行为，然后可以对其进行计划和操作。

有关最新参数，请参考 AWS::GroundStation::MissionProfile CloudFormation 资源类型

1. 命名你的任务简介。这使您可以快速了解其在系统中的使用情况。例如，satellite-narrowband-emergency-operations如果有单独的窄带载波用于satellite-wideband-narrowband-nominal紧急操作，则可能有- operations 和 a。
2. 设置您的跟踪配置。
3. 设置最小可行接触时长。这使您可以筛选潜在的联系人以满足您的任务需求。
4. 设置用于streamsKmsRole在传输过程中加密数据的streamsKmsKey和。这用于所有 AWS Ground Station 代理数据流。
5. 设置您的数据流。使用您在上一步中创建的配置创建数据流以匹配您的载波信号。

6. [可选] 设置通过前和通过后的接触持续时间（秒）。这用于分别在联系人之前和之后发出每个联系人的事件。请参阅利用事件 AWS Ground Station 实现自动化了解更多信息。
7. [可选] 您可以将标签与任务资料相关联。这些可以用来帮助以编程方式区分您的任务配置文件。

您可以参考任务配置文件配置示例，仅查看一些潜在的配置。

了解后续步骤

既然您已经拥有了机载卫星和有效的任务配置文件，就可以安排联系人并与卫星通信了。 AWS Ground Station

您可以通过以下方式之一安排联系：

- 控制AWS Ground Station 台。
- AWS CLI 的 r eserve-contact 命令。
- AWS 软件开发工具包。 ReserveContactAPI。

有关如何 AWS Ground Station 跟踪卫星轨迹以及如何使用这些信息的信息，请参阅了解如何 AWS Ground Station 使用卫星星历数据。

AWS Ground Station 维护了许多预配置的 AWS CloudFormation 模板，以便更轻松地开始使用该服务。任务配置文件配置示例有关 AWS Ground Station 如何使用的示例，请参阅。

处理数字中频数据或从中提供给您的解调和解码数据 AWS Ground Station 将取决于您的具体用例。以下博客文章可以帮助您了解一些可用的选项：

- 使用 AWS Ground Station Amazon S3 数据传输进行自动地球观测（及其关联 GitHub 存储库 awslabs/ aws-groundstation-eos-pipeline）
- 使用虚拟化卫星地面段 AWS
- 地球观测使用 AWS Ground Station：如何指导
- 使用 d@@ igiF AWS Ground Station WideBand 和 Amphinicy Blink SDR（以及相关的存储库 aws-samples/）构建高吞吐量的卫星数据下行链路架构 GitHub aws-groundstation-wbdigif-snpp

AWS Ground Station 地点

AWS Ground Station 提供全球地面站网络，紧邻我们的 AWS 基础设施区域全球网络。您可以从任何支持的 AWS 区域配置对这些位置的使用。这包括传输数据的 AWS 区域。

查找地面站位置的 AWS 区域

AWS Ground Station 全球网络包括实际位置不在与之相连的 AWS 地区的地面站地点。您可以通过 AWS 开发工具包 ListGroundStation 响应检索您有权访问的地面站列表。地面站位置的完整列表如下所示，更多的地面站位置即将推出。请参阅入职指南，为您的卫星添加或修改场地许可。

Ground Station 名称	Ground Station 位置	AWS 区域名称	AWS 区域代码	备注
阿拉斯加 1	美国阿拉斯加州	美国西部（俄勒冈州）	us-west-2	实际不位于某个 AWS 区域
巴林 1	巴林	中东（巴林）	me-south-1	

Ground Station 名称	Ground Station 位置	AWS 区域名称	AWS 区域代码	备注
开普敦 1	南非开普敦	非洲（开普敦）	af-south-1	
Dubbo 1	澳大利亚达博	亚太地区（悉尼）	ap-southeast-2	实际不位于某个 AWS 区域
夏威夷 1	美国夏威夷	美国西部（俄勒冈州）	us-west-2	实际不位于某个 AWS 区域
爱尔兰 1	爱尔兰	欧洲地区（爱尔兰）	eu-west-1	
俄亥俄州 1	美国俄亥俄州	美国东部（俄亥俄州）	us-east-2	
俄勒冈 1	美国俄勒冈州	美国西部（俄勒冈州）	us-west-2	
蓬塔阿雷纳斯 1	智利蓬塔阿雷纳斯	南美洲（圣保罗）	sa-east-1	实际不位于某个 AWS 区域
首尔 1	韩国首尔	亚太地区（首尔）	ap-northeast-2	
新加坡 1	新加坡	亚太地区（新加坡）	ap-southeast-1	
斯德哥尔摩 1	斯德哥尔摩，瑞典	欧洲地区（斯德哥尔摩）	eu-north-1	

AWS Ground Station 支持的 AWS 区域

您可以通过 AWS 软件开发工具包或支持的 AWS 区域的 AWS Ground Station 控制台提供数据和配置联系人。您可以在终端节点和配额处查看支持的区域及其关联的AWS Ground Station 终端节点。

数字双胞胎可用性

使用 AWS Ground Station 数字双胞胎功能适用于所有可用的 AW AWS Ground Station S 区域。数字双胞胎地面站是生产地面站的精确副本，Ground Station 名称的前缀修改为 “数字双胞胎”。例如，“Digital Twin Ohio 1” 是一个数字双胞胎地面站，它是 “俄亥俄一号” 生产地面站的精确副本。

AWS Ground Station 网站掩码

每个 AWS Ground Station 天线位置都有相关的站点掩码。当指向某些方向（通常靠近地平线）时，这些掩码会阻止天线在该位置的发射或接收信号。掩码可以考虑：

- 天线周围的地理地形特征 — 例如，这包括山脉或建筑物之类的东西，它们会阻挡射频 (RF) 信号或阻止传输。
- 射频干扰 (RFI) — 这会影响接收（影响到 AWS Ground Station 天线的下行链路信号的外部 RFI 源）和传输（AWS Ground Station 天线传输的射频信号，对外部接收器产生不利影响）的能力。
- 法律授权 — 在每个地区运营 AWS Ground Station 的本地站点授权可能包括特定限制，例如传输的最小仰角。

这些站点掩码可能会随时间而变化。例如，可以在天线位置附近建造新建筑，RFI 来源可能会发生变化，或者可能会根据不同的限制续订法律授权。根据保密协议 (NDA)，您可以获得 AWS Ground Station 网站掩码。

客户专用口罩

除了每个站点的 AWS Ground Station 站点掩码外，您可能还会有额外的口罩，因为您自己的法律授权受到限制，无法与特定区域的卫星通信。可以在 AWS Ground Station 中配置此类掩码，以确保使用 AWS Ground Station 与这些卫星通信时的合规性。 case-by-case请联系 AWS Ground Station 团队了解详情。

现场口罩对可用联系时间的影响

有两种站点掩码：上行链路（传输）站点掩码和下行链路（接收）站点掩码。

使用该 ListContacts 操作列出可用的联系时间时，AWS Ground Station 将根据您的卫星何时升至下行链路掩码下方并设置在下行屏蔽下方返回能见时间。可用的联系时间基于此下行链路掩码可见性窗口。这样可以确保当您的卫星低于下行链路掩码时，您不会预留时间。

即使任务配置文件在数据流边缘包含天线上行链路配置，上行链接站点掩码也不会应用于可用的联络时间。这使您可以将所有可用的联系时间用于下行链路，即使由于上行链路站点掩码的原因，上行链路可能在部分时间内不可用。但是，在为卫星联络预留的部分或全部时间内，可能无法传输上行链路信号。在安排上行链路传输时，您有责任考虑所提供的上行链路掩码。

相对于天线位置的上行链路站点掩码，不可用于上行链路的部分联络会因联络期间卫星轨迹的不同而有所不同。在上行链路和下行链路站点掩码相似的区域，此持续时间通常会很短。在其他区域，上行链路掩码可能明显高于下行链路站点掩码，这可能会导致上行链路在很大部分甚至全部联络期间无法使用。即使部分预留时间不可用于上行链路，也会向您收取全部联系时间。

AWS Ground Station 网站能力

为了简化您的体验，请为一种天线类型 AWS Ground Station 确定一组常用功能，然后将多根天线部署到地面站位置。部分入门步骤可确保您的卫星与特定位置的天线类型兼容。当您预订联系人时，您可以间接确定所使用的天线类型。这样可以确保无论使用哪种天线，您在特定地面站位置的体验都会随着时间的推移保持不变。由于各种环境问题（例如现场的天气），您的联系人的具体表现会有所不同。

目前，所有站点都支持以下功能：

> **Note**
>
> 除非另有说明，否则下表中的每一行都表示独立的通信路径。存在重复行是为了反映我们的多渠道功能，允许同时使用多条通信路径。

能力类型	频率范围	带宽范围	Polarization	公用名	备注
天线下行链路	7750-8500 MHz	50-400 MHz	RHCP	X 波段宽带下行链路	此功能需要使用代AWS Ground Station 理。 阿拉斯加 1 或蓬塔阿雷纳斯 1 不支持此功能。
天线下行链路	7750-8500 MHz	50-400 MHz	RHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	RHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	RHCP		

能力类型	频率范围	带宽范围	Polarization	公用名	备注
天线下行链路	7750-8500 MHz	50-400 MHz	RHCP		每个位置的 MHz 每次极化总带宽不得超过 400。 所有使用的频率范围必须不重叠。
天线下行链路	7750-8500 MHz	50-400 MHz	LHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	LHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	LHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	LHCP		
天线下行链路	7750-8500 MHz	50-400 MHz	LHCP		
天线下行链路	2200-2290 MHz	最多 40 MHz	RHCP	S 波段下行链路	一次只能使用一种极化
天线下行链路	2200-2290 MHz	最多 40 MHz	LHCP		
天线下行链路	7750-8500 MHz	最多 40 MHz	RHCP	X 波段窄带下行链路	一次只能使用一种极化
天线下行链路	7750-8500 MHz	最多 40 MHz	LHCP		
天线上行链路	2025-2110 MHz	最多 40 MHz	RHCP	S 波段上行链路	一次只能使用一种极化
天线上行链路	2025-2110 MHz	最多 40 MHz	LHCP		EIRP 20-50 dBW
antenna-uplink-echo	2025-2110 MHz	2 MHz	RHCP	上行链路回声	匹配天线上行链路限制

能力类型	频率范围	带宽范围	Polarization	公用名	备注
antenna-uplink-echo	2025-2110 MHz	2 MHz	LHCP		
antenna-downlink-demod-decode	7750-8500 MHz	最多 500 MHz	RHCP	X 波段解调和解码下行链路	
antenna-downlink-demod-decode	7750-8500 MHz	最多 500 MHz	LHCP		
跟踪	不适用	不适用	不适用	不适用	Support 支持自动跟踪和节目跟踪

* RHCP = 右手圆极化，LHCP = 左手圆极化。有关极化的更多信息，请参见圆极化。

了解如何 AWS Ground Station 使用卫星星历数据

一个星历，或多个星历，是一种提供天体轨迹的文件或数据结构。从历史上看，该文件仅涉及表格数据，但逐渐演变成为指示航天器轨迹的各种数据文件。

AWS Ground Station 使用星历数据来确定您的卫星何时有联系人可用，并正确命令 AWS Ground Station 网络中的天线指向您的卫星。默认情况下，如果您的卫星已分配了 NOR AWS Ground Station AD ID，则无需执行任何操作即可提供星历表。

主题

默认星历数据

默认情况下， AWS Ground Station 使用来自 Space -Track 的公开数据，无需执行任何操作即可 AWS Ground Station 提供这些默认星历表。这些星历表是与卫星的 NORAD ID 关联的两行元素集 (TLEs)。所有默认星历的优先级均为 0。因此，它们总是会被通过星历 API 上传的任何未过期的自定义星历覆盖，其优先级必须始终为 1 或更高。

没有 NORAD ID 的卫星必须将自定义星历数据上传到。 AWS Ground Station例如，刚刚发射或故意从太空轨道目录中省略的卫星将没有 NORAD ID，需要上传自定义星历表。有关提供自定义星历的更多信息，请参阅：提供自定义星历数据。

提供自定义星历数据

> **Important**
>
> 星历 API 当前处于预览状态

星历 API 的访问权限仅在需要时提供。<#################### aws-groundstation@amazon.com#> AWS Ground Station 将星历视为个性化使用数据。如果您使用此可选功能，AWS 将使用您的星历数据来提供故障排除支持。

概览

Ephemeris API 允许将自定义星历上传到卫星上以供卫星使用。 AWS Ground Station 这些星历表覆盖了 Space-Track 中的默认星历表（参见：）。 默认星历数据我们支持以 Orbit Ephemeris 消息 (OEM) 和双行元素 (TLE) 格式接收星历数据。

上传自定义星历表可以提高跟踪质量，在没有 Spac e-Track 星历表的情况下处理早期操作，并考虑机动。 AWS Ground Station

> **Note**
>
> 在为卫星分配卫星目录编号之前提供自定义星历时，可以使用 00000 作为 TLE 的卫星目录编号字段，使用 000 作为 TLE 或 OEM 元数据的国际标号字段的发射编号部分（例如，24000A 表示 2024 年发射的飞行器）。
> 有关格式的更多信息 TLEs，请参阅双行元素集。有关格式的更多信息 OEMs，请参阅OEM 星历格式。

OEM 星历格式

AWS Ground Station 根据 CCSDS 标准处理 OEM 客户提供的星历表，但有一些额外的限制。OEM 文件应采用 KVN 格式。下表概述了 OEM 中的不同字段以及 AWS Ground Station 与 CCSDS 标准的区别。

Section	字段	需要 CCSDS	AWS Ground Station 必需的	备注
标题	CCSDS_OEM_VERS	支持	是	所需值：2.0
	COMMENT	否	否	
	分类	否	否	
	创建日期	支持	是	

Section	字段	需要 CCSDS	AWS Ground Station 必需的	备注
	鼻祖	支持	是	
	消息_ID	否	否	
元数据	META_START	支持	是	
	COMMENT	否	否	
	对象名	支持	是	
	对象_ID	支持	是	
	中心/名称	支持	是	必填值：地球
	REF_FRAME	支持	是	可接受的值：EME2000, ITRF2 000
	REF_FRAME _EPOCH	否	不支持*	不需要，因为接受的 REF_ FRAMEs 有一个隐式的时代
	时间系统	支持	是	必填值：世界标准时间
	开始时间	支持	是	
	可用开始时间	否	否	
	USEABLE_S TOP_TIME	否	否	
	停止时间	支持	是	

Section	字段	需要 CCSDS	AWS Ground Station 必需的	备注
	插值	否	是	必需，因此 AWS Ground Station 可以为触点生成精确的指向角度。
	插值度	否	是	必需，因此 AWS Ground Station 可以为触点生成精确的指向角度。
	META_STOP	支持	是	
数据	X 形	支持	是	代表于 km
	Y	支持	是	代表于 km
	Z	支持	是	代表于 km
	X_DOT	支持	是	代表于 km/s
	Y_DOT	支持	是	代表于 km/s
	Z_DOT	支持	是	代表于 km/s
	X_DDOT	否	否	代表于 km/s^2
	Y_DDOT	否	否	代表于 km/s^2
	Z_DDOT	否	否	代表于 km/s^2
协方差矩阵	协方差起点	否	否	
	EPOCH	否	否	

Section	字段	需要 CCSDS	AWS Ground Station 必需的	备注
	COV_REF_FRAME	否	否	
	协方差停止	否	否	

* 如果提供的 OEM 中包含任何 AWS Ground Station 不支持的行，OEM 将无法通过验证。

与 CCSDS 标准的重要差异是： AWS Ground Station

- 必须是 CCSDS_OEM_VERS。`2.0`
- REF_FRAME 必须是或`EME2000`。`ITRF2000`
- 不支持 REF_FRAME_EPOCH。 AWS Ground Station
- 必须为 CENTER_NAME。`Earth`
- TIME_SYSTEM 必须是。`UTC`
- CPE 都需要插值和插值_度。 AWS Ground Station

KVN 格式的 OEM 星历示例

以下是 JPSS-1 公共广播公司卫星的 KVN 格式的 OEM 星历的截断示例。

```
CCSDS_OEM_VERS = 2.0

COMMENT Orbit data are consistent with planetary ephemeris DE-430

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

META_START
OBJECT_NAME          = J1
OBJECT_ID            = 2017-073A
CENTER_NAME          = Earth
REF_FRAME            = EME2000
TIME_SYSTEM          = UTC
START_TIME           = 2024-07-22T00:00:00.000000
```

```
STOP_TIME            = 2024-07-22T00:06:00.000000
INTERPOLATION        = Lagrange
INTERPOLATION_DEGREE = 5
META_STOP

2024-07-22T00:00:00.000000   5.905147360000000e+02  -1.860082793999999e+03
  -6.944807075000000e+03  -5.784245796000000e+00   4.347501391999999e+00
 -1.657256863000000e+00
2024-07-22T00:01:00.000000   2.425572045154201e+02  -1.595860765983339e+03
  -7.030938457373539e+03  -5.810660250794190e+00   4.457103652219009e+00
 -1.212889340333023e+00
2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
  -7.090262617183503e+03  -5.814973972202444e+00   4.549739160042560e+00
 -7.639633689161465e-01
2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
  -7.122556683227951e+03  -5.797176562437553e+00   4.625064829516728e+00
 -3.121687831090774e-01
2024-07-22T00:04:00.000000  -8.015427368657785e+02  -7.709137891269565e+02
  -7.127699477194810e+03  -5.757338007808417e+00   4.682800822515077e+00
 1.407953645161997e-01
2024-07-22T00:05:00.000000  -1.145240083085062e+03  -4.886583601179489e+02
  -7.105671911254255e+03  -5.695608435738609e+00   4.722731329786999e+00
 5.932259682105052e-01
2024-07-22T00:06:00.000000  -1.484582479061495e+03  -2.045451985605701e+02
  -7.056557069672793e+03  -5.612218005854990e+00   4.744705579872771e+00
 1.043421397392599e+00
```

创建自定义星历表

可以使用 AWS Ground Station API 中的 CreateEphemeris 操作创建自定义星历。此操作将使用请求正文或指定 S3 存储桶中的数据上传星历。

请务必注意，上传星历会将星历设置为 VALIDATING 并启动异步工作流程，该工作流程将验证您的星历并生成可能联络。只有当星历通过此工作流程并成为 ENABLED 后，才会将其用于联络。您应该轮DescribeEphemeris询星历状态或使用 CloudWatch 事件来跟踪星历的状态变化。

要对无效的星历进行故障排除，请参阅：对无效的星历进行故障排除

示例：通过 API 创建双行元素 (TLE) 集星历表

和 CLI 可用于通过调用将双行元素 (TLE) 集星历表 AWS Ground Station 上传到。 AWS SDKs CreateEphemeris此星历将用于代替卫星的默认星历数据（参见默认星历数据）。此示例展示了如何使用适用于 Python 的AWS SDK (Boto3) 来执行此操作。

TLE 集是一个 JSON 格式的对象，它将一个或多个串在 TLEs 一起以构造连续的轨迹。TLE 集合 TLEs 中的必须形成一个连续的集合，我们可以用它来构造轨迹（即 TLE 集合 TLEs 中没有时间间隔）。下面显示了一个 TLE 集示例：

```
# 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 集合 TLEs 中的时间范围必须完全匹配才能成为有效的连续轨迹。

TLE 集可以通过 bot AWS Ground Station o3 客户端上传，如下所示：

```
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)
                }
            }
        ]
      }
    })
```

此调用将返回一个 ephemeriSid，将来可用于引用星历表。例如，我们可以使用上面调用中提供的 EphemeriSid来轮询星历的状态：

```
client.describe_ephemeris(ephemerisId=tle_ephemeris_id['ephemerisId'])
```

下面提供了 DescribeEphemeris 操作的响应示例

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

```
}
```

建议轮询DescribeEphemeris路径或使用 CloudWatch 事件来跟踪上传的星历表的状态，因为它必须经过异步验证工作流程，然后才能设置为并可用于安排ENABLED和执行联系人。

请注意，在上面的示例中，TLE 集合 TLEs 中所有的 NORAD ID 必须与25994在 SpaceTrack 数据库中为您的卫星分配的 NORAD ID 相匹配。

示例：从 S3 存储桶上传星历数据

也可以通过指向 S3 存储桶和对象密钥直接从 S3 存储桶上传星历文件。 AWS Ground Station 将代表您取回对象。有关静态数据加密的信息，请参阅：AWS Ground Station 的静态数据加密 AWS Ground Station

以下是从 S3 存储桶上传 OEM 星历文件的示例

```
s3_oem_ephemeris_id = ground_station_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",
            }
        }
    })
```

以下是为上一个示例代码块中上传的 OEM 星历调用 DescribeEphemeris 操作返回的数据示例。

```
{
  "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"
```

```
      }
    }
  }
}
```

示例：使用客户提供的星历表 AWS Ground Station

有关使用客户提供的星历表的更多详细说明，请参阅将客户提供的星历表与（及其 AWS Ground Station关联的存储库 aws-samples/）一起 AWS Ground Station使用 GitHub aws-groundstation-cpe

了解使用的是哪种星历

星历具有优先级、到期时间和启用标记。它们共同决定哪个星历可用于卫星。每颗卫星只能有一个星历处于活动状态。

将要使用的星历是已启用的具有最高优先级的星历，其到期时间在未来。优先级值越大表示优先级越高。返回的可用联系时间ListContacts基于此星历。如果多个 ENABLED 的星历具有相同的优先级，则将使用最近创建或更新的星历。

> **Note**
>
> AWS Ground Station 对每颗卫星ENABLED客户提供的星历表数量有服务配额（参见：Service Quotas）。要在达到此限额后上传星历数据，请删除（使用 DeleteEphemeris）或禁用（使用 UpdateEphemeris）优先级最低/最早创建的客户提供的星历。

如果尚未创建星历表，或者没有星历表具有ENABLED状态，则 AWS Ground Station 将使用卫星的默认星历表（来自 Space-Track）（如果有）。此默认星历的优先级为 0。

新星历表对先前安排的接触的影响

使用 DescribeContact API 通过返回活动可见时间来查看新星历对先前安排的联系人的影响。

在上传新星历之前安排的联系人将保留最初安排的联系时间，而天线跟踪将使用有效的星历表。如果基于活动星历的航天器位置与先前的星历有很大不同，则由于航天器在发射/接收站点掩码之外运行，这可能会导致卫星与天线的接触时间缩短。因此，我们建议您在上传与之前的星历表大不相同的新星历后，取消并重新安排未来的联系时间。借助 DescribeContact API，您可以通过将您的预定接触与返回的接触进行比较，来确定由于航天器在发射/接收站点掩码之外运行 `startTime` 而endTime无法使用的部分。`visibilityStartTime` `visibilityEndTime`如果您选择取消并重新安排将来的联系

人，则联系时间范围在可见时间范围之外的时间不得超过 30 秒。如果取消联系的时间太近，则取消的联系可能会产生费用。有关取消联系的更多信息，请参阅：G round Station FAQs。

获取卫星的当前星历

可以通过调用或操作来检索 AWS Ground Station 特定卫星当前使用的星历表。GetSatelliteListSatellites这两种方法都将返回当前使用的星历的元数据。上传到默认星历表的自定义星历表和默认星历表的星历表元数据有所不同。 AWS Ground Station

默认星历将仅包含 source 和 epoch 字段。epoch这是从 Sp ace-Track 中提取的双线元素集的时代，它目前正用于计算卫星的轨迹。

自定义星历的 "CUSTOMER_PROVIDED" 为 source 值，且在 ephemerisId 字段中包含唯一标识符。此唯一标识符可用于通过 DescribeEphemeris 操作查询星历。如果在上传时 AWS Ground Station 通过操作为星历分配了名称，则会返回一个可选字name段。CreateEphemeris

值得注意的是，星历表是通过动态更新的， AWS Ground Station 因此返回的数据只是调用 API 时使用的星历的快照。

使用默认星历的卫星示例返回 GetSatellite

```
{
    "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
    }
}
```

使用自定义星历的卫星示例 GetSatellite

```
{
      "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"
      }
  }
```

恢复为默认星历数据

当你上传自定义星历数据时，它将覆盖该特定卫星使用的默认星历表。 AWS Ground Station AWS Ground Station 在没有当前启用、未过期的客户提供的星历表可供使用之前，不会再次使用默认的星历表。 AWS Ground Station 也不会列出超过当前客户提供的星历到期时间的联系人，即使在该到期时间之后有默认的星历表可用。

要恢复到默认的 Sp ace-Track 星历表，你需要执行以下操作之一：

- 删除（使用 DeleteEphemeris）或禁用（使用 UpdateEphemeris）所有客户提供的已启用星历。您可以使用 ListEphemerides 列出客户提供的卫星的星历。
- 等待客户提供的所有现有星历到期。

您可以通过调用 GetSatellite 并验证卫星当前星历的 `source` 是否为 `SPACE_TRACK` 来确认是否正在使用默认星历。有关默认默认星历数据星历表的更多信息，请参阅。

使用数据流

AWS Ground Station 使用节点和边缘关系来构造数据流，以便对数据进行流处理。每个节点都由一个描述其预期处理的配置表示。为了说明这个概念，可以考虑将数据流设置`antenna-downlink`为a。`s3-recording`该`antenna-downlink`节点表示根据配置中定义的参数对射频频谱进行模拟到数字的转换。`s3-recording`表示一个计算节点，它将接收传入的数据并将其存储在您的 S3 存储桶中。生成的数据流是根据您的规格将数字化的 RF 数据异步传输到 S3 存储桶。

在任务配置文件中，您可以创建许多数据流来满足您的需求。以下各节介绍如何设置要与之配合使用的其他 AWS 资源， AWS Ground Station 并提供了构建数据流的建议。有关每个节点行为方式的详细信息，包括将其视为源节点还是目标节点，请参阅使用 AWS Ground Station 配置。

主题

AWS Ground Station 数据平面接口

所选数据流的结果数据结构取决于数据流的来源。这些格式的详细信息将在卫星装载期间提供给您。以下汇总了每种类型的数据流所使用的格式。

- 天线下行链路
  - （带宽小于 54MHz）数据以 VITA-49 信号数据/IP 格式数据包的形式传送。
  - （带宽 greater-than-or-equal-到 54MHz）数据作为 2 AWS Ground Station 类数据包传送。
- antenna-downlink-demod-decode
  - 数据以Demodulated/Decoded Data/IP格式化数据包的形式传送。
- 天线上行链路
  - 数据必须作为 VITA-49 信号数据/IP 格式数据包传送。
- antenna-uplink-echo
  - 数据以 VITA-49 信号数据/IP 格式数据包的形式传送。

使用跨区域数据传输

AWS Ground Station 跨区域数据传输功能使您可以灵活地将数据从天线发送到任何 AWS Ground Station 支持的 AWS 区域。这意味着，您可以在单个 AWS 区域维护您的基础设施，并安排任何 AWS Ground Station AWS Ground Station 地点已上线区域的联系时间。

在 Amazon S3 存储桶中接收您的联系人数据时，所有 AWS Ground Station 支持的区域目前都支持跨区域数据传输。 AWS Ground Station 将为您管理所有配送方面。

所有地区均可 EC2 使用 AWS Ground Station 代理向亚马逊跨 antenna-to-destination区域传输数据。此设置不需要唯一的配置或批准。

默认情况下，在下述区域中， EC2 可以使用数据流终端节点向亚马逊跨区域传输数据*。 antenna-to-destination

- 美国东部（俄亥俄）区域 (us-east-2) 到美国西部（俄勒冈）区域 (us-west-2)
- 美国西部（俄勒冈）区域 (us-west-2) 至美国东部（俄亥俄）区域 (us-east-2)

要使用跨区域数据传输到 Amazon EC2 实例，必须在您当前的 AWS 区域中创建数据流终端节点，并且dataflow-endpoint-config必须指定相同的区域。

下表汇总了前面的信息，详细说明了跨区域数据传输的支持区域和交付方式。

收款方法	天线区域	收货区域
亚马逊 S3 数据传输	全部上线 AWS Ground Station AWS Ground Station 地点	所有AWS Ground Station 区域
AWS Ground Station Amazon 上的代理 EC2	全部上线 AWS Ground Station AWS Ground Station 地点	所有AWS Ground Station 区域
Amazon 上的 Dataflow 终端节点* EC2	美国东部（俄亥俄州）区域 (us-east-2)	美国西部（俄勒冈州）区域 (us-west-2)
	美国西部（俄勒冈州）区域 (us-west-2)	美国东部（俄亥俄州）区域 (us-east-2)

*未列出的其他 antenna-to-destination地区需要特殊的 Amazon EC2 和软件设置。
请<## aws-groundstation@amazon.com> 联系我们，获取入职说明。

设置和配置 Amazon S3

您可以使用 Amazon S3 存储桶通过以下方式 AWS Ground Station接收下行链路信号。要创建目标 s3 录制配置，您必须能够指定 Amazon S3 存储桶和授权 AWS Ground Station 向存储桶写入文件的 IAM 角色。

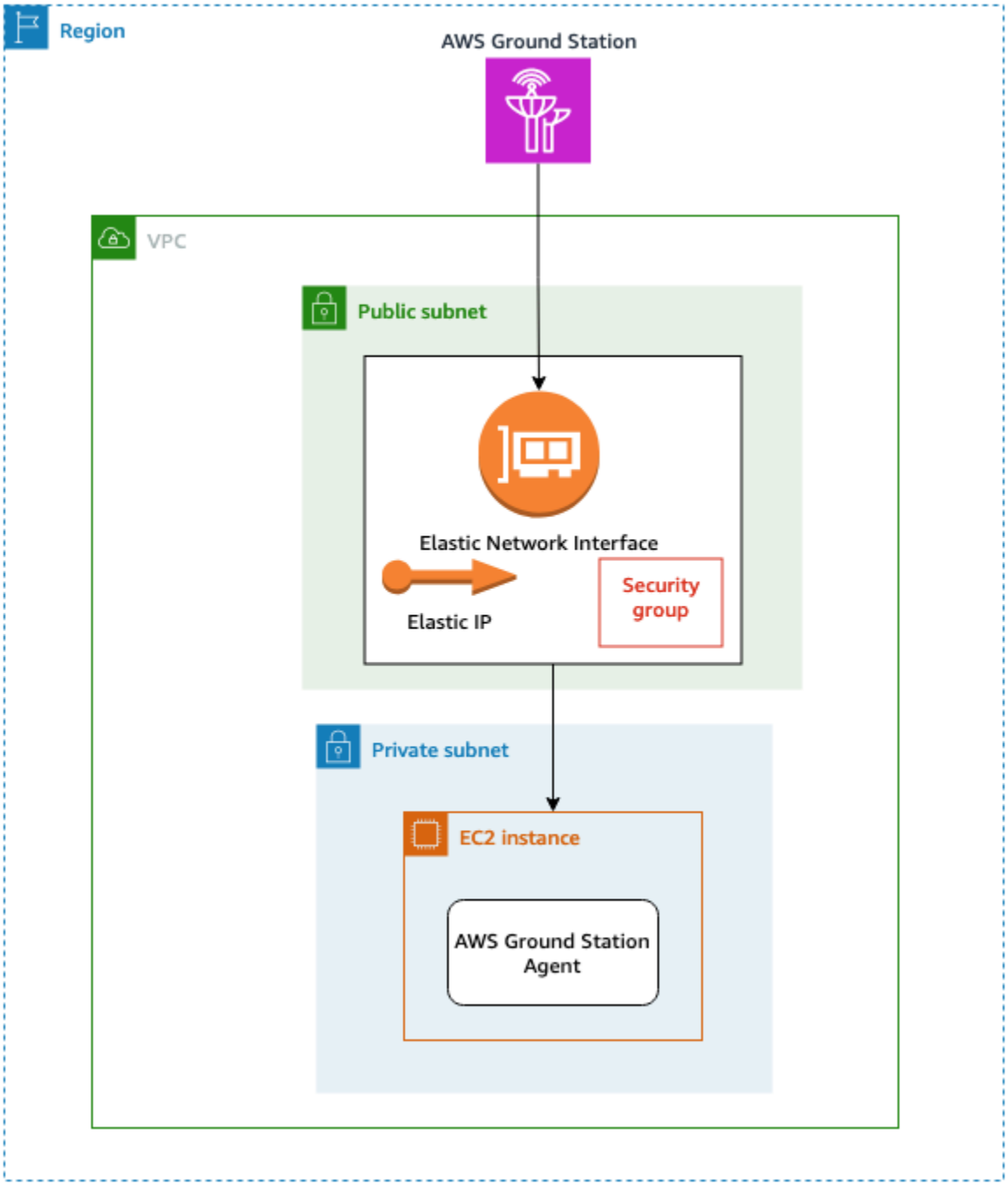
亚马逊 S3 录音 Config有关 Amazon S3 存储桶、IAM 角色或 AWS Ground Station 配置创建的限制，请参阅。

设置和配置 Amazon VPC

设置 VPC 的完整指南超出了本指南的范围。要深入了解，请参阅 Amazon VPC 用户指南。

在本节中，将介绍您的 Amazon EC2 和 Dataflow 终端节点如何存在于 VPC 中。 AWS Ground Station 不支持给定数据流的多个传送点——预计每个数据流都会终止到一个接收器。 EC2 正如我们所期望的那样，只有一个 EC2 接收器，该配置不是多可用区冗余的。有关使用您的 VPC 的完整示例，请参阅任务配置文件配置示例。

使用 AWS Ground Station 代理配置 VPC



您的卫星数据将提供给靠近天线的 AWS Ground Station 代理实例。 AWS Ground Station 代理将对您的数据进行条带化处理，然后使用您提供的 AWS KMS 密钥对其进行加密。每个条带都通过 AWS 网络主干从源天线发送到您的亚马逊 EC2 弹性 IP (EIP)。数据通过附带的 Amazon EC2 弹性网络接口 (ENI) 到达您的 EC2 实例。进入您的 EC2实例后，安装的 AWS Ground Station 代理将解密您的数据并执行正向错误校正 (FEC) 以恢复所有丢失的数据，然后将其转发到您在设置中指定的 IP 和端口。

下表列出了在设置 VPC 以进行 AWS Ground Station 代理交付时的独特设置注意事项。

安全组-建议您设置一个仅用于 AWS Ground Station 流量的安全组。此安全组应允许在您的 Dataflow 端点组中指定的相同端口范围上的 UDP 入口流量。 AWS Ground Station 维护 AWS 管理的前缀列表，将您的权限仅限于 AWS Ground Station IP 地址。有关如何替换您的部署区域的详细信息，请参阅 AWS 托管前缀列表。PrefixListId

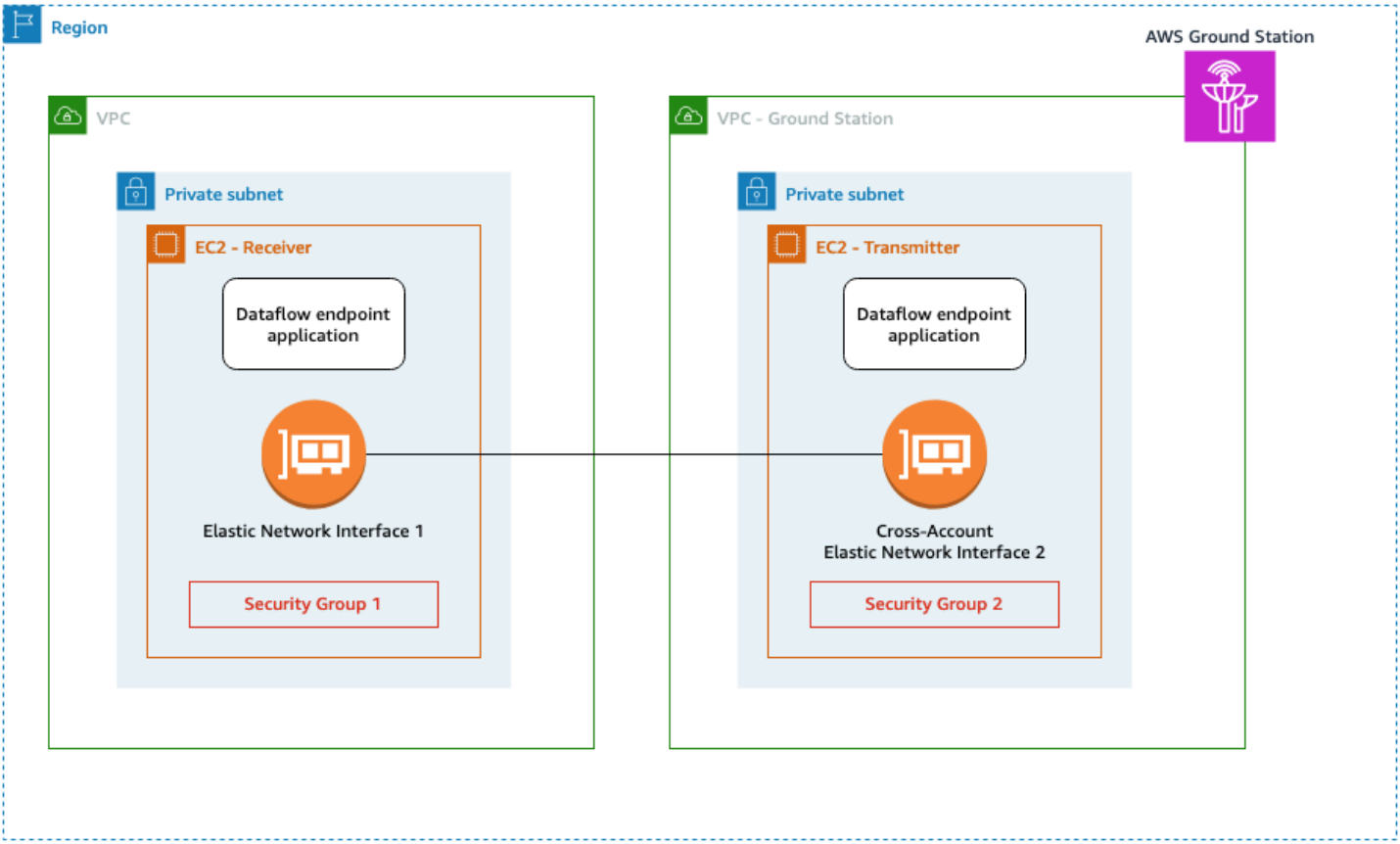
弹性网络接口 (ENI)-您需要将上述安全组与此 ENI 关联并将其放置在您的公有子网中。

以下 CloudFormation 模板演示如何创建本节中描述的基础架构。

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

InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
    VpcId:YourVpcId
    SecurityGroupIngress:
      # Add additional items here.
      - IpProtocol: udp
        FromPort: your-port-start-range
        ToPort: your-port-end-range
        PrefixListIds:
          - PrefixListId: com.amazonaws.global.groundstation
        Description: "Allow AWS Ground Station Downlink ingress."

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

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

```
    NetworkInterfaceId:
      Ref: InstanceNetworkInterface
```

带有数据流终端节点的 VPC 配置



您的卫星数据将提供给靠近天线的 Dataflow 端点应用程序实例。然后，数据将通过跨账户 Amazon EC2 弹性网络接口 (ENI) 从其拥有的 AWS Ground Station VPC 发送。然后，数据通过附加到您的 Amazon EC2 EC2 实例的 ENI 到达您的实例。然后，安装的 dataflow 端点应用程序会将其转发到您在设置中指定的 IP 和端口。对于上行链路连接，则会出现相反的情况。

下表列出了为数据流终端节点传输设置 VPC 时的独特设置注意事项。

IAM 角色-IAM 角色是 Dataflow 终端节点的一部分，未显示在图表中。用于创建跨账户 ENI 并将其附加到 A AWS Ground Station mazon EC2 实例的 IAM 角色。

安全组 1-此安全组附加到 ENI，该弹性网卡将与您账户中的 Amazon EC2 实例相关联。它需要允许来自安全组 2 的 UDP 流量通过您的dataflow-endpoint-group中指定的端口。

弹性网络接口 (ENI) 1-您需要将安全组 1 与此 ENI 关联并将其放置在子网中。

子网-您需要确保您的账户中每个数据流至少有一个可用的 IP 地址可用于 Amazon EC2 实例。有关子网大小的更多详细信息，请参阅 “子网 CIDR 块”

安全组 2-在 Dataflow 端点中引用了此安全组。此安全组将附加到用于 AWS Ground Station 将数据存入您的账户的 ENI。

区域-有关跨区域连接支持的区域的更多信息，请参阅使用跨区域数据传输。

以下 CloudFormation 模板演示如何创建本节中描述的基础架构。

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

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

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

ReceiverSubnet:
  Type: AWS::EC2::Subnet
  Properties:
```

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

设置和配置 Amazon EC2

要通过 AWS Ground Station 代理或数据流终端节点同步交付 VITA-49Signal/IP data or VITA-49 Extension data/IP，需要正确配置您的 Amazon EC2 实例。根据您的特定需求，您可以直接在同一个实例上执行前端 (FE) 处理器或软件定义无线电 (SDR)，或者可能需要使用其他 EC2实例。FE 或 SDR 的选择和安装超出了本用户指南的范围。有关特定数据格式的更多信息，请参阅AWS Ground Station 数据平面接口。

有关我们的服务条款的信息，请参阅AWS 服务条款。

提供的通用软件

AWS Ground Station 提供了常用软件来简化您的 Amazon EC2 实例的设置。

AWS Ground Station 代理人

AWS Ground Station 代理接收数字中频 (digiF) 下行链路数据并输出解密后的数据，从而实现以下功能：

- digiF 下行链路能力从 40 MHz 到 400 MHz 的带宽不等。
- 向网络上的 AWS 任何公共 IP AWS （弹性 IP）传输高速率、低抖动 digiF 数据。
- 使用前向纠错 (FEC) 实现可靠的数据传输。
- 使用客户托管 AWS KMS 密钥进行加密来保护数据传输。

有关更多信息，请参阅《AWS Ground Station 代理用户指南》。

数据流端点应用程序

一种网络应用程序，用于在 AWS Ground Station 天线位置和您的 AWS Ground Station Amazon EC2 实例之间发送和接收数据。它可用于数据的上行链路和下行链路。

软件定义无线电 (SDR)

一种软件定义的无线电 (SDR)，可用于调制/解调用于与卫星通信的信号。

AWS Ground Station Amazon 机器映像 (AMIs)

为了缩短这些安装的构建和配置时间， AWS Ground Station 还提供了预先配置 AMIs的选项。 AMIs 带有 dataflow 端点的网络应用程序和软件定义无线电 (SDR) 将在您的登录完成后提供给您的账户。它们可以在亚马逊 EC2 控制台中通过在私有 Amazon M achine Images (AMIs) 中搜索地面站来找到。with AMIs A AWS Ground Station gent 是公开的，可以在亚马逊 EC2 控制台中通过在公共亚马逊机器映像 (AMIs) 中搜索地面站来找到。

处理联系人

您可以使用所选语言的 AWS Ground Station 控制台或 AWS SDK 输入卫星数据、识别天线位置 AWS CLI、通信和安排所选卫星的天线时间。您最晚可以在联系开始前 15 分钟查看、取消和重新安排联系人预订*。此外，如果您使用预留分钟数定价模式，则可以查看 AWS Ground Station 预留分钟数定价计划的详细信息。

AWS Ground Station 支持跨区域数据传输。作为您选择的任务配置文件一部分的数据流终端节点配置决定了将数据传输到哪个区域。有关使用跨区域数据传输的更多信息，请参阅使用跨区域数据传输。

要安排联络，必须配置您的资源。如果您尚未配置资源，请参阅开始使用。ReserveContact被调用时， AWS Ground Station 拍摄任务配置文件和配置资源的快照，以便在接触通行证期间使用。使用UpdateMissionProfile和对这些资源所做的更改UpdateConfig APIs 不会反映在更新前保留的联系人中。如果您需要将资源更改应用于已安排的联系人，则必须先使用取消该联系 CancelContact，然后使用重新安排该联系。ReserveContact

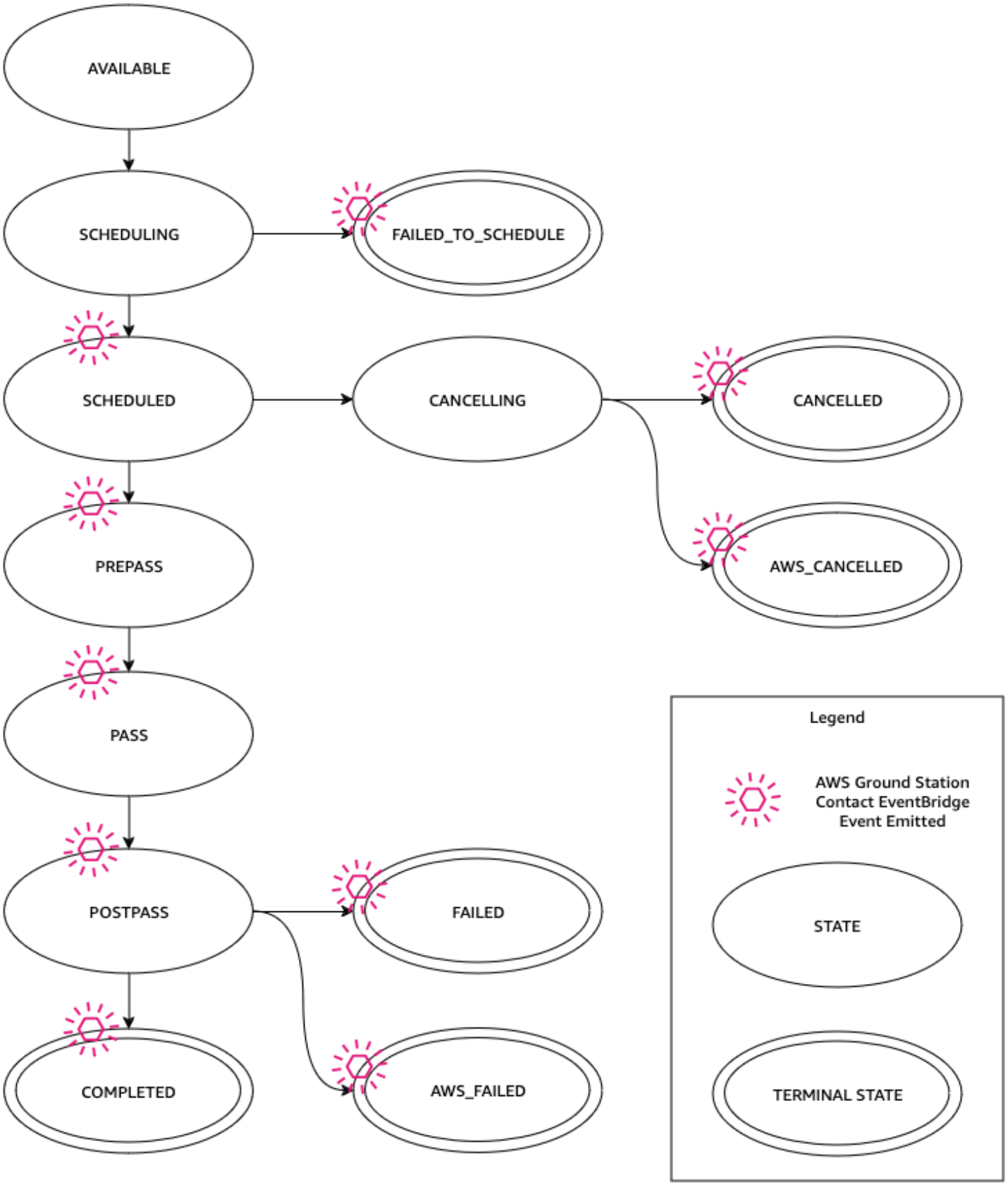
* 如果取消联系的时间太近，则取消的联系可能会产生费用。有关取消联系的更多信息，请参阅：Ground Station FAQs。

主题

- 了解联系人生命周期

了解联系人生命周期

了解联系人生命周期有助于确定如何配置自动化以及如何进行故障排除。下图显示了 AWS Ground Station 联系人生命周期以及生命周期中发出的 Event Bridge 事件。请务必注意，“已完成”、“失败”、“FAILED_TO_SCHEDULE”、“已取消” 和 “已终止” AWS_CANCELLED 状态。 AWS_FAILED联系人不会从终端状态过渡出来。AWS Ground Station 联系人状态有关每种状态所表示的内容的详细信息，请参阅。


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

AWS Ground Station 联系人状态

通过 AWS Ground Station 联系人的状态，可以深入了解该联系人在给定时间发生了什么。

联系人状态

以下是联络可以具有的状态列表：

- AVAILABLE：联络可供预订。
- SCHEDULING：联络正在安排中。
- SCHEDULED：已成功安排联络。
- FAILED_TO_SCHEDULE：联络安排失败。
- PREPASS：联络即将开始，正在准备资源。
- PASS：联络当前正在执行，并且正在与卫星通信。
- POSTPASS：通信已完成，正在清理所使用的资源。
- 已完成-联系已完成，没有错误。
- 失败-由于您的资源配置存在问题，联系失败。
- AWS_FAILED-由于 AWS Ground Station 服务出现问题，联系失败。
- CANCELLING：正在取消联络。
- AWS_CANCELLED-该 AWS Ground Station 服务取消了联系。天线或场地维护以及星历漂移是何时可能发生这种情况的例子。
- 已取消-联系已被您取消。

使用 AWS Ground Station 数字双胞胎功能

的数字双胞胎功能为您 AWS Ground Station 提供了一个可以测试和集成卫星任务管理和指挥与控制软件的环境。数字双胞胎功能允许您在不使用生产天线容量的情况下测试调度、验证配置和正确的错误处理。通过测试与数字双胞胎功能的 AWS Ground Station 集成，您可以增强对系统顺利管理卫星运行的能力的信心。它还允许您在 AWS Ground Station APIs不使用生产容量或需要频谱许可的情况下进行测试。

要开始使用，请关注机载卫星，申请加入数字双胞胎功能。一旦您的卫星登上数字双胞胎功能，您就可以安排与数字双胞胎地面站的联系。您可以通过 AWS 开发工具包ListGroundStations响应检索您有权访问的地面站列表。数字双胞胎地面站是中列出的地面站的精确副本，AWS Ground Station 地点Ground Station 名称的修改前缀为 “数字双胞胎”。这包括他们的天线功能和元数据，包括但不限于站点掩码和实际的 GPS 坐标。目前，数字双胞胎功能不支持数据传输，如中所述使用数据流。

一旦上线，数字双胞胎功能就会发出与生产服务相同的 Amazon EventBridge 事件和 API 响应，如中所述。利用事件 AWS Ground Station 实现自动化这些事件将允许您微调配置和数据流端点组。

通过以下方式了解监控 AWS Ground Station

监控是保持 AWS Ground Station可靠性、可用性和性能的重要环节。AWS 提供以下监控工具 AWS Ground Station，供您监视、报告问题并在适当时自动采取措施。

- Amazon EventBridge Events 提供近乎实时的系统事件流，这些事件描述了 AWS 资源的变化。EventBridge 事件支持事件驱动的自动计算，因为您可以编写规则来监视某些事件，并在这些事件发生时在其他 AWS 服务中触发自动操作。有关 EventBridge 活动的更多信息，请参阅 Amazon EventBridge Events 用户指南。
- AWS CloudTrail捕获由您的账户或代表您的 AWS 账户进行的 API 调用和相关事件，并将日志文件传输到您指定的 Amazon S3 存储桶。您可以识别哪些用户和帐户拨打了电话 AWS、发出呼叫的源 IP 地址以及呼叫发生的时间。有关的更多信息 AWS CloudTrail，请参阅《AWS CloudTrail 用户指南》。
- 使用时，Amazon CloudWatch Metrics 会捕获您的预定联系指标 AWS Ground Station。CloudWatch Metrics 使您能够根据自己的频道、极化和卫星 ID 分析数据，以识别信号强度和联系人中的错误。有关更多信息，请参阅使用 Amazon CloudWatch 指标。
- AWS 用户通知服务 可用于设置交付渠道以获取有关 AWS Ground Station 事件的通知。当事件与指定的规则匹配时，会收到通知。您可以通过多个渠道接收事件通知，包括电子邮件、聊天应用程序中的 Amazon Q 开发者版聊天通知或 AWS Console Mobile Application 推送通知。您还可以在 AWS 控制台通知中心查看通知。 用户通知服务 支持聚合，这可以减少您在特定事件期间收到的通知数量。

使用以下主题监控 AWS Ground Station。

主题

- 利用事件 AWS Ground Station 实现自动化
- 使用记录 AWS Ground Station API 调用 AWS CloudTrail
- 通过 Amazon 查看指标 CloudWatch

利用事件 AWS Ground Station 实现自动化

> **Note**
>
> 本文档自始至终都使用 “事件” 一词。 CloudWatch 事件和 EventBridge 是相同的底层服务和 API。可以使用任何一种服务构建用于匹配传入事件并将事件路由至目标进行处理的规则。

事件使您能够实现 AWS 服务自动化，并自动响应系统事件，例如应用程序可用性问题或资源更改。来自 AWS 服务的事件近乎实时地交付。您可以编写简单规则来指示您关注的事件，并指示要在事件匹配规则时执行的自动化操作。一些可以自动触发的操作包括：

- 调用函数 AWS Lambda
- 调用 Amazon EC2 运行命令
- 将事件中继到 Amazon Kinesis Data Streams
- 激活 AWS Step Functions 状态机
- 通知 Amazon SNS 主题或 Amazon SQS 队列

将事件与一起使用的一些示例 AWS Ground Station 包括：

- 调用 Lambda 函数，根据事件状态自动启动和停止 A EC2 mazon 实例。
- 每当联络变更状态时发布到 Amazon SNS 主题。这些主题可以设置为在开始或结束联络时发送电子邮件通知。

有关更多信息，请参阅 Amazon Ev EventBridge ents 用户指南。

AWS Ground Station 事件类型

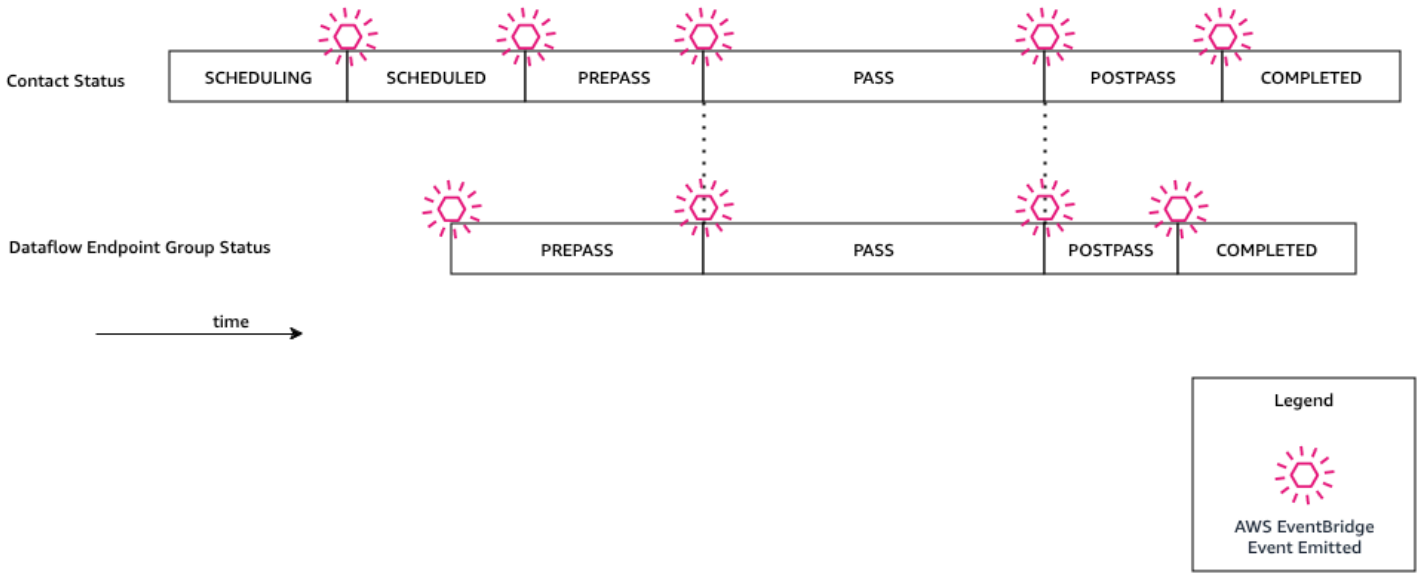
> **Note**
>
> AWS Ground Station 生成的所有事件都将“aws.groundstation”作为“来源”的值。

AWS Ground Station 发出与状态更改相关的事件，以支持您自定义自动化。目前， AWS Ground Station 支持联系人状态更改事件、数据流端点组更改事件和星历状态更改事件。以下各节提供了有关每种类型的详细信息。

联系活动时间表

AWS Ground Station 当您的联系人更改状态时会发出事件。有关这些状态变化是什么以及各州本身含义的更多信息，请参阅了解联系人生命周期。您的联系人中使用的任何 dataflow 端点组都有一组独立的事件，这些事件也会发出。在同一时间段内，我们还会为你的数据流终端节点组发出事件。在设置任务配置文件和数据流端点组时，您可以配置通过前和通过后事件的精确时间。

下图显示了名义联系人及其关联的 Dataflow 端点组的状态和发出的事件。



Ground Station 联络状态变更

如果您想在即将到来的联系人更改状态时执行特定操作，则可以设置规则来自动执行此操作。当您想要接收有关联络状态变更的通知时，这很有用。如果您想更改收到这些事件的时间，可以修改任务配置文件 contactPrePassDurationSeconds和 contactPostPassDurationSeconds。事件将发送到计划联络的区域。

下面提供了一个示例事件。

```
{
    "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"
    }
}
```

`contactStatus` 可使用的值在 the section called “AWS Ground Station 联系人状态” 中定义。

Ground Station 数据流终端节点组状态变更

如果要在数据流终端节点组正用于接收数据时执行操作，则可以设置 规则以自动执行此操作。这将允许您执行不同的操作来响应数据流终端节点组状态的不断变更的状况。如果您想更改收到这些事件的时间，请使用具有不同contactPrePassDurationSeconds和的 dataflow 端点组。contactPostPassDurationSeconds此事件将发送到数据流终端节点组的区域。

下面提供了一个示例。

```
{
    "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"
    }
}
```

dataflowEndpointGroupState 的可能状态包括 PREPASS、PASS、POSTPASS 和 COMPLETED。

星历事件

Ground Station Ephemeris 状态变更

如果想要在星历变更状态时执行操作，则可设置规则以自动执行此操作。这允许您根据星历的变化状态执行不同的操作。例如，您可以在星历完成验证且现在已 ENABLED 时执行操作。此事件的通知将发送到上传星历的区域。

下面提供了一个示例。

```
{
    "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 的可能状态包括 ENABLED、VALIDATING、INVALID、ERROR、DISABLED 和 EXPIRED

使用记录 AWS Ground Station API 调用 AWS CloudTrail

AWS Ground Station 与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在中执行的操作的记录 AWS Ground Station。 CloudTrail 将所有 API 调用捕获 AWS Ground Station 为事件。捕获的调用包括来自 AWS Ground Station 控制台的调用和对 AWS Ground Station API 操作的代码调用。如果您创建跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括的事件 AWS Ground Station。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的事件历史记录中查看最新的事件。使用收集的信息 CloudTrail，您可以确定向哪个请求发出 AWS Ground Station、发出请求的 IP 地址、谁发出了请求、何时发出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅AWS CloudTrail 用户指南。

AWS Ground Station 中的信息 CloudTrail

CloudTrail 在您创建 AWS 账户时已在您的账户上启用。当活动发生在中时 AWS Ground Station，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在自己的 AWS 账户中查看、搜索和下载最近发生的事件。有关更多信息，请参阅使用事件历史记录查看 CloudTrail 事件。

要持续记录您 AWS 账户中的事件，包括的事件 AWS Ground Station，请创建跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。默认情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和从多个账户接收 CloudTrail 日志文件

所有 AWS Ground Station 操作均由 API 参考记录 CloudTrail 并记录在 AWS Ground Station API 参考中。例如，调用CancelContact和ListConfigs操作会在 CloudTrail 日志文件中生成条目。ReserveContact

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

了解 AWS Ground Station 日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定顺序出现。

以下示例显示了演示该`ReserveContact`操作的 CloudTrail 日志条目。

示例 ReserveContact：

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:sts::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2019-05-15T21:11:59Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::123456789012:role/Alice",
                "accountId": "123456789012",
                "userName": "Alice"
            }
        }
    },
```

```
    "eventTime": "2019-05-15T21:14:37Z",
    "eventSource": "groundstation.amazonaws.com",
    "eventName": "ReserveContact",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "Mozilla/5.0 Gecko/20100101 Firefox/123.0",
    "requestParameters": {
        "satelliteArn":
 "arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555",
        "groundStation": "Ohio 1",
        "startTime": 1558356107,
        "missionProfileArn": "arn:aws:groundstation:us-east-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555",
        "endTime": 1558356886
    },
    "responseElements": {
        "contactId": "11111111-2222-3333-4444-555555555555"
    },
    "requestID": "11111111-2222-3333-4444-555555555555",
    "eventID": "11111111-2222-3333-4444-555555555555",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "11111111-2222-3333-4444-555555555555"
}
```


通过 Amazon 查看指标 CloudWatch

在联系期间，AWS Ground Station 自动捕获数据并将其发送到以 CloudWatch 供分析。您的数据可以在亚马逊 CloudWatch 控制台中查看。有关访问和 CloudWatch 指标的更多信息，请参阅使用 Amazon CloudWatch 指标。

AWS Ground Station 指标和维度

有哪些指标可用？

以下指标可从中获得 AWS Ground Station。

> **Note**
>
> 发出的具体指标取决于所使用的 AWS Ground Station 功能。根据您的配置，只能发出以下指标的子集。

指标	指标维度	描述
AzimuthAngle	SatelliteId	天线的方位角。真北为 0 度，东为 90 度。 单位：度
BitErrorRate	信道、极化、 SatelliteId	在给定数量的位传输中的位上误差率。位误差是由噪声、失真或干扰引起的 单位：每单位时间的位误差数量
BlockErrorRate	信道、极化、 SatelliteId	给定数量的接收块中块的误差率。块误差是由干扰引起的。 单位：错误的块数/总块数
CarrierFrequencyRecovery_Cn0	类别、Config、 SatelliteId	单位带宽的载波噪声密度比。 单位：分贝赫兹 (db-Hz)
CarrierFrequencyRecovery_Locked	类别、Config、 SatelliteId	解调器载波频率恢复环路锁定时设置为 1，解锁时设置为 0。 单位：无

指标	指标维度	描述
CarrierFrequencyRecovery_OffsetFrequency_Hz	类别、Config、 SatelliteId	预估信号中心频率与理想中心频率之间的偏差。这是由多普勒频移和航天器与天线系统之间的局部振荡器偏移引起的。 单位：赫兹 (Hz)
ElevationAngle	SatelliteId	天线的仰角。地平线为 0 度，天顶为 90 度。 单位：度
Es/N0	信道、极化、 SatelliteId	每个符号的能量与噪声功率谱密度的比率。 单位：分贝 (dB)
ReceivedPower	极化， SatelliteId	解调器/解码器中测量的信号强度。 单位：分贝毫瓦 (dBm)
SymbolTimingRecovery_ErrorVectorMagnitude	类别、Config、 SatelliteId	接收符号和理想星座点之间的误差矢量幅度。 单位：百分比

指标	指标维度	描述
SymbolTimingRecovery_Locked	类别、Config、 SatelliteId	解调器符号时序恢复环路锁定时设置为 1，解锁时设置为 0。 单位：无
SymbolTimingRecovery_OffsetSymbolRate	类别、Config、 SatelliteId	预估符号速率和理想信号符号速率之间的偏差。这是由多普勒频移和航天器与天线系统之间的局部振荡器偏移引起的。 单位：符号/秒

尺寸是用来做什么用的 AWS Ground Station？

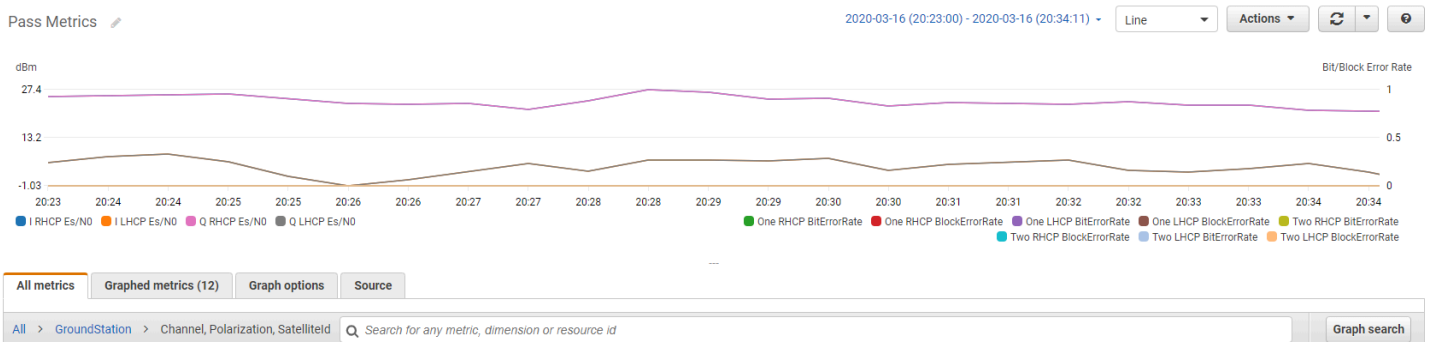
您可以使用以下维度筛选 AWS Ground Station 数据。

维度	描述
Category	解调或解码。
Channel	每次联络的通道包括 1、2、I（相中）和 Q（正交）。
Config	天线下行链路解码解码配置 arn。
Polarization	每次联络的偏振包括 LHCP（左手圆偏振）或 RHCP（右手圆偏振）。
SatelliteId	卫星 ID 包含用于联络的卫星的 ARN。

查看 指标

查看图形化的指标时，请务必注意，聚合窗口决定了指标的显示方式。联络中的每个指标都可以在收到数据后 3 小时内显示为每秒的数据。在 3 小时后，您的数据将按每分钟的数据按 CloudWatch 指标汇总。如果您需要查看每秒数据测量的指标，建议您在收到数据后的 3 小时内查看数据，或者将其保留在 CloudWatch 指标之外。有关 CloudWatch 留存率的更多信息，请参阅 Amazon CloudWatch 概念-指标保留率。

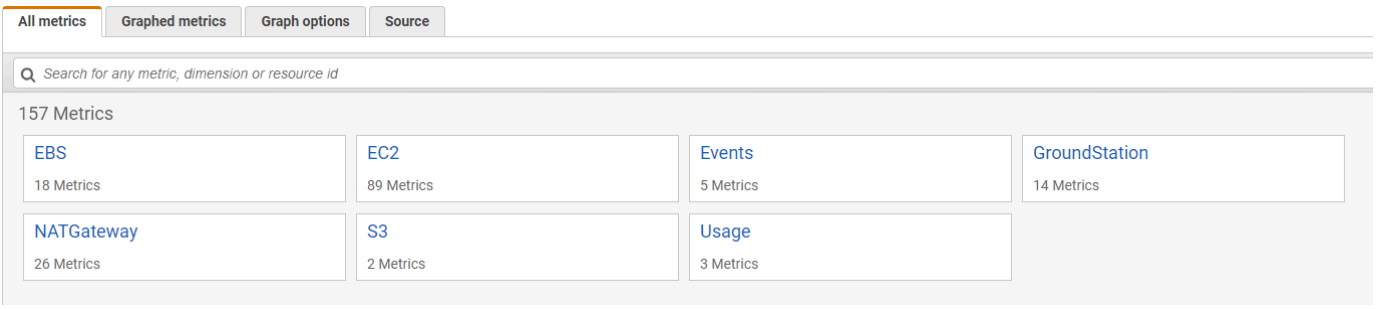
此外，在前 60 秒内捕获的任何数据都不会包含足够的信息来生成有意义的指标，并且很可能不会显示。为了查看有意义的指标，建议在 60 秒后查看您的数据。



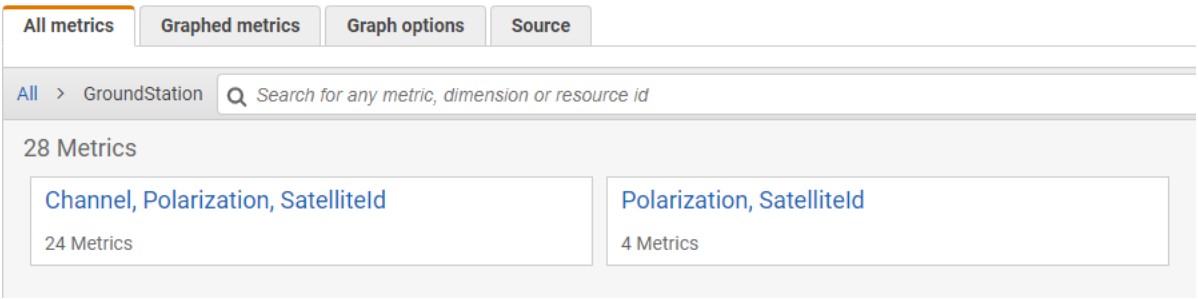
有关在中绘制 AWS Ground Station 指标图表的更多信息 CloudWatch，请参阅绘制指标图表。

使用控制台查看指标

1. 打开 CloudWatch 管理控制台。
2. 在导航窗格中，选择指标。
3. 选择 GroundStation 命名空间。



4. 选择所需的指标维度（例如，信道、极化等 SatelliteId）。



5. All metrics 选项卡显示命名空间中该维度的所有指标。您可执行以下操作：

   a. 要对表进行排序，请使用列标题。

   b. 要绘制指标的图表，请选中与该指标关联的复选框。要选择所有指标，请选中表格标题行中的复选框。

   c. 要按资源进行筛选，请选择资源 ID，然后选择 Add to search。

   d. 要按指标进行筛选，请选择指标名称，然后选择 Add to search。

要查看指标，请使用以下方法 AWS CLI

1. 确保 AWS CLI 已安装。有关安装的信息 AWS CLI，请参阅安装 AWS CLI 版本 2。

2. 使用 CloudWatch CLI 的get-metric-data方法生成一个文件，该文件可以修改以指定您感兴趣的指标，然后用于查询这些指标。

   为此，请运行以下命令：`aws cloudwatch get-metric-data --generate-cli-skeleton`。这将生成类似于以下内容的输出：

```
{
    "MetricDataQueries": [
        {
            "Id": "",
            "MetricStat": {
                "Metric": {
                    "Namespace": "",
                    "MetricName": "",
                    "Dimensions": [
                        {
                            "Name": "",
                            "Value": ""
                        }
```

```
                    ]
                 },
                 "Period": 0,
                 "Stat": "",
                 "Unit": "Seconds"
              },
              "Expression": "",
              "Label": "",
              "ReturnData": true,
              "Period": 0,
              "AccountId": ""
           } ],
        "StartTime": "1970-01-01T00:00:00",
        "EndTime": "1970-01-01T00:00:00",
        "NextToken": "",
        "ScanBy": "TimestampDescending",
        "MaxDatapoints": 0,
        "LabelOptions": {
           "Timezone": ""
        }
     }
```

3. 通过运行列出可用 CloudWatch 指标`aws cloudwatch list-metrics`。

   如果您最近使用过 AWS Ground Station，则该方法应返回包含以下条目的输出：

```
     ...
     {
        "Namespace": "AWS/GroundStation",
        "MetricName": "ReceivedPower",
        "Dimensions": [
           {
              "Name": "Polarization",
              "Value": "LHCP"
           },
           {
              "Name": "SatelliteId",
              "Value": "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-
bbbb-cccc-dddd-eeeeeeeeeeee"
           }
        ]
     },
```

```
        ...
```

> **Note**
>
> 由于限制 CloudWatch，如果自上次使用以来已超过 2 周 AWS Ground Station，则需要手动检查可用指标表，以便在指标命名空间中查找指标名称和维度。AWS/GroundStation有关 CloudWatch 限制的更多信息，请参阅：查看可用指标

4. 修改您在步骤 2 中创建的 JSON 文件，使其与步骤 3 中的所需值以及您的指标Polarization中的所需值相匹配。SatelliteId另外，请务必更新StartTime、和EndTime值以匹配您的联系人。例如：

```
        {
           "MetricDataQueries": [
              {
                 "Id": "receivedPowerExample",
                 "MetricStat": {
                    "Metric": {
                       "Namespace": "AWS/GroundStation",
                       "MetricName": "ReceivedPower",
                       "Dimensions": [
                          {
                             "Name": "SatelliteId",
                             "Value":
 "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-bbbb-cccc-dddd-
eeeeeeeeeeee"
                          },
                          {
                             "Name": "Polarization",
                             "Value": "RHCP"
                          }
                       ]
                    },
                    "Period": 300,
                    "Stat": "Maximum",
                    "Unit": "None"
                 },
                 "Label": "ReceivedPowerExample",
                 "ReturnData": true
              }
```

```
            ],
            "StartTime": "2024-02-08T00:00:00",
            "EndTime": "2024-04-09T00:00:00"
         }
```

> **Note**
>
> AWS Ground Station 每 1 到 60 秒发布一次指标，具体取决于指标。如果该Period字段的值小于指标的发布周期，则不会返回指标。

5. `aws cloudwatch get-metric-data`使用在前面步骤中创建的配置文件运行。下面提供了一个示例。

```
aws cloudwatch get-metric-data --cli-input-json file://
<nameOfConfigurationFileCreatedInStep2>.json
```

将向指标提供来自联络的时间戳。下面提供了 AWS Ground Station 指标的输出示例。

```
{
   "MetricDataResults": [
      {
         "Id": "receivedPowerExample",
         "Label": "ReceivedPowerExample",
         "Timestamps": [
            "2024-04-08T18:35:00+00:00",
            "2024-04-08T18:30:00+00:00",
            "2024-04-08T18:25:00+00:00"
         ],
         "Values": [
            -33.30191555023193,
            -31.46100273132324,
            -32.13915576934814
         ],
         "StatusCode": "Complete"
      }
   ],
   "Messages": []
}
```

安全性 AWS Ground Station

云安全 AWS 是重中之重。作为 AWS 客户，您将受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。AWS 提供专用的安全工具和特性，以帮助实现安全目标。这些工具和功能包括网络安全性、配置管理、访问控制和数据安全性。

使用时 AWS Ground Station，我们建议您遵循行业最佳实践并实施 end-to-end加密。AW APIs S 为您提供集成加密和数据保护的功能。有关 AWS 安全的更多信息，请参阅 AWS 安全简介白皮书。

使用以下主题了解如何保护您的资源。

主题

- Identity and Access Management AWS Ground Station
- AWS 的托管策略 AWS Ground Station
- 在 Ground Station 中使用与服务相关的角色
- 静态数据加密 AWS Ground Station
- 传输期间的数据加密 AWS Ground Station

Identity and Access Management AWS Ground Station

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（拥有权限）使用 AWS Ground Station 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

主题

- 受众
- 使用身份进行身份验证
- 使用策略管理访问
- 如何 AWS Ground Station 与 IAM 配合使用
- 基于身份的策略示例 AWS Ground Station
- 对 AWS Ground Station 身份和访问进行故障排除

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您所做的工作 AWS Ground Station。

服务用户-如果您使用 AWS Ground Station 服务完成工作，则管理员会为您提供所需的凭证和权限。当你使用更多 AWS Ground Station 功能来完成工作时，你可能需要额外的权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 AWS Ground Station中的特征，请参阅 对 AWS Ground Station 身份和访问进行故障排除。

服务管理员-如果您负责公司的 AWS Ground Station 资源，则可能拥有完全访问权限 AWS Ground Station。您的工作是确定您的服务用户应访问哪些 AWS Ground Station 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要详细了解您的公司如何将 IAM 与配合使用 AWS Ground Station，请参阅如何 AWS Ground Station 与 IAM 配合使用。

IAM 管理员：如果您是 IAM 管理员，您可能希望了解如何编写策略以管理对 AWS Ground Station的访问权限的详细信息。要查看您可以在 IAM 中使用的 AWS Ground Station 基于身份的策略示例，请参阅。基于身份的策略示例 AWS Ground Station

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证（登录 AWS）。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。 AWS IAM Identity Center （IAM Identity Center）用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当你使用联合访问 AWS 时，你就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》中的如何登录到您 AWS 账户的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅《IAM 用户指南》中的用于签署 API 请求的AWS 签名版本 4。

无论使用何种身份验证方法，您都可能需要提供其他安全信息。例如， AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的多重身份验证和《IAM 用户指南》中的 IAM 中的AWS 多重身份验证。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务 和资源。此身份被称为 AWS 账户 root 用户，使用您创建帐户时使用的电子邮件地址和密码登录即可访问该身份。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求您以根用户身份登录的任务的完整列表，请参阅 IAM 用户指南中的需要根用户凭证的任务。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity C enter 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。 AWS Directory Service当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅 AWS IAM Identity Center 用户指南中的什么是 IAM Identity Center？。

IAM 用户和群组

I AM 用户是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定的使用场景需要长期凭证以及 IAM 用户，建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的对于需要长期凭证的用例，应在需要时更新访问密钥。

IAM 组是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可以拥有一个名为的群组，IAMAdmins并向该群组授予管理 IAM 资源的权限。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的 IAM 用户的使用案例。

IAM 角色

I AM 角色是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。要在中临时担任 IAM 角色 AWS Management Console，您可以从用户切换到 IAM 角色（控制台）。您可

以通过调用 AWS CLI 或 AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的代入角色的方法。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问：要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关用于联合身份验证的角色的信息，请参阅《IAM 用户指南》中的针对第三方身份提供商创建角色（联合身份验证）。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的权限集。
- 临时 IAM 用户权限：IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。
- 跨账户存取：您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅 IAM 用户指南中的 IAM 中的跨账户资源访问。
- 跨服务访问 — 有些 AWS 服务 使用其他 AWS 服务服务中的功能。例如，当您在服务中拨打电话时，该服务通常会在 Amazon 中运行应用程序 EC2 或在 Amazon S3 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。
  - 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。
  - 服务相关角色-服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。
- 在 A@@ mazon 上运行的应用程序 EC2 — 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这比在 EC2 实例中存储访问密钥更可取。要为 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建一个附加到该实例的实例配置文件。实例配置文件包含角色并允许在 EC2 实例上运行的程序获得临时证书。有关更多信息，请参阅 IAM 用户指南中的使用 IAM 角色向在 A mazon EC2 实例上运行的应用程序授予权限。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS ，当与身份或资源关联时，它会定义其权限。 AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的JSON 策略概览。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户托管策略定义自定义 IAM 权限。

基于身份的策略可以进一步归类为内联策略或托管式策略。内联策略直接嵌入单个用户、组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管策略和内联策略之间进行选择，请参阅《IAM 用户指南》中的在托管策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACLs)

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持的服务示例 ACLs。 AWS WAF要了解更多信息 ACLs，请参阅《亚马逊简单存储服务开发者指南》中的访问控制列表 (ACL) 概述。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- 权限边界：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅IAM 用户指南中的 IAM 实体的权限边界。
- 服务控制策略 (SCPs)- SCPs 是指定组织或组织单位 (OU) 的最大权限的 JSON 策略 AWS Organizations。 AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果您启用组织中的所有功能，则可以将服务控制策略 (SCPs) 应用于您的任何或所有帐户。SCP 限制成员账户中的实体（包括每个 AWS 账户根用户实体）的权限。有关 Organization SCPs s 和的更多信息，请参阅《AWS Organizations 用户指南》中的服务控制策略。
- 资源控制策略 (RCPs) — RCPs 是 JSON 策略，您可以使用它来设置账户中资源的最大可用权限，而无需更新附加到您拥有的每个资源的 IAM 策略。RCP 限制成员账户中资源的权限，并可能影响身份（包括身份）的有效权限 AWS 账户根用户，无论这些身份是否属于您的组织。有关 Organizations 的更多信息 RCPs，包括 AWS 服务 该支持的列表 RCPs，请参阅《AWS Organizations 用户指南》中的资源控制策略 (RCPs)。
- 会话策略：会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅 IAM 用户指南中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的策略评估逻辑。

如何 AWS Ground Station 与 IAM 配合使用

在使用 IAM 管理访问权限之前 AWS Ground Station，请先了解哪些可用的 IAM 功能 AWS Ground Station。

您可以搭配使用的 IAM 功能 AWS Ground Station

IAM 特征	AWS Ground Station 支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键（特定于服务）	是
ACLs	否
ABAC（策略中的标签）	是
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	是

要全面了解 AWS Ground Station 以及其他 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的AWS 服务。

基于身份的策略 AWS Ground Station

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

基于身份的策略示例 AWS Ground Station

要查看 AWS Ground Station 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS Ground Station

内部基于资源的政策 AWS Ground Station

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

要启用跨账户访问，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的主体。将跨账户主体添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源处于不同位置时 AWS 账户，可信账户中的 IAM 管理员还必须向委托人实体（用户或角色）授予访问资源的权限。他们通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

的政策行动 AWS Ground Station

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看 AWS Ground Station 操作列表，请参阅《服务授权参考》 AWS Ground Station中定义的操作。

正在执行的策略操作在操作前 AWS Ground Station 使用以下前缀：

```
groundstation
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

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

要查看 AWS Ground Station 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS Ground Station

的政策资源 AWS Ground Station

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

`Resource` JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

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

要查看 AWS Ground Station 资源类型及其列表 ARNs，请参阅《服务授权参考》 AWS Ground Station中定义的资源。要了解可以在哪些操作中指定每个资源的 ARN，请参阅 AWS Ground Station 定义的操作。

要查看 AWS Ground Station 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS Ground Station

的策略条件密钥 AWS Ground Station

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看 AWS Ground Station 条件键列表，请参阅《服务授权参考》 AWS Ground Station中的条件密钥。要了解可以使用条件键的操作和资源，请参阅由定义的操作 AWS Ground Station。

要查看 AWS Ground Station 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS Ground Station

ACLs in AWS Ground Station

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

ABAC with AWS Ground Station

支持 ABAC（策略中的标签）：是

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以将标签附加到 IAM 实体（用户或角色）和许多 AWS 资源。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

将临时凭证与配合使用 AWS Ground Station

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IA M 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

的跨服务主体权限 AWS Ground Station

支持转发访问会话（FAS）：是

当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详细信息，请参阅转发访问会话。

AWS Ground Station的服务角色

支持服务角色：否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

> **Warning**
>
> 更改服务角色的权限可能会中断 AWS Ground Station 功能。只有在 AWS Ground Station 提供操作指导时才编辑服务角色。

的服务相关角色 AWS Ground Station

支持服务相关角色：是

服务相关角色是一种与服务相关联的 AWS 服务服务角色。服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理服务相关角色的详细信息，请参阅能够与 IAM 搭配使用的AWS 服务。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

基于身份的策略示例 AWS Ground Station

默认情况下，用户和角色没有创建或修改 AWS Ground Station 资源的权限。他们也无法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略（控制台）。

有关由 AWS Ground Station定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《服务授权参考》 AWS Ground Station中的操作、资源和条件密钥。 ARNs

主题

- 策略最佳实践

- 使用 AWS Ground Station 控制台
- 允许用户查看他们自己的权限

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 AWS Ground Station 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定的方式使用的，则也可以使用条件来授予对服务操作的访问权限 AWS 服务，例如 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅 IAM 用户指南中的 IAM 中的安全最佳实操。

使用 AWS Ground Station 控制台

要访问 AWS Ground Station 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 AWS Ground Station 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保用户和角色仍然可以使用 AWS Ground Station 控制台，还需要将 AWS Ground Station *ConsoleAccess*或*ReadOnly* AWS 托管策略附加到实体。有关更多信息，请参阅《IAM 用户指南》中的为用户添加权限。

允许用户查看他们自己的权限

该示例说明了您如何创建策略，以允许 IAM 用户查看附加到其用户身份的内联和托管式策略。此策略包括在控制台上或使用 AWS CLI 或 AWS API 以编程方式完成此操作的权限。

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

对 AWS Ground Station 身份和访问进行故障排除

使用以下信息来帮助您诊断和修复在使用 AWS Ground Station 和 IAM 时可能遇到的常见问题。

主题

我无权在以下位置执行操作 AWS Ground Station

如果您收到错误提示，指明您无权执行某个操作，则必须更新策略以允许执行该操作。

当 `mateojackson` IAM 用户尝试使用控制台查看有关虚构 *`my-example-widget`* 资源的详细信息，但不拥有虚构 `groundstation:`*`GetWidget`* 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 groundstation:GetWidget on resource: my-example-widget
```

在此情况下，必须更新 `mateojackson` 用户的策略，以允许使用 `groundstation:`*`GetWidget`* 操作访问 *`my-example-widget`* 资源。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我无权执行 iam：PassRole

如果您收到一个错误，表明您无权执行 `iam:PassRole` 操作，则必须更新策略以允许您将角色传递给 AWS Ground Station。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 `marymajor` 的 IAM 用户尝试使用控制台在 AWS Ground Station中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 `iam:PassRole` 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人 AWS 账户 访问我的 AWS Ground Station 资源

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表 (ACLs) 的服务，您可以使用这些策略向人们授予访问您的资源的权限。

要了解更多信息，请参阅以下内容：

- 要了解是否 AWS Ground Station 支持这些功能，请参阅如何 AWS Ground Station 与 IAM 配合使用。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户 ，请参阅 IAM 用户指南中的向您拥有 AWS 账户 的另一个 IAM 用户提供访问权限。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅 IAM 用户指南中的向第三方提供访问权限。 AWS 账户
- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问之间的差别，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

AWS 的托管策略 AWS Ground Station

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住， AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。 AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

AWS 托管策略： AWSGroundStationAgentInstancePolicy

您可以将 AWSGroundStationAgentInstancePolicy 策略附加到 IAM 身份。

该策略向 AWS Ground Station 代理授予对您的 Amazon EC2 实例的权限，允许该实例在 Ground Station 联系期间发送和接收数据。本策略中的所有权限均来自 Ground Station 服务。

权限详细信息

该策略包含以下权限。

- groundstation— 允许数据流端点实例调用 Ground Station 代理。 APIs

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

AWS 托管策略： AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy

您无法附加 AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy 到您的 IAM 实体。此策略附加到允许代表您执行操作 AWS Ground Station 的服务相关角色。有关更多信息，请参阅使用服务相关角色。

此策略授予 AWS Ground Station 允许查找公共 IPv4 地址的 EC2 权限。

权限详细信息

该策略包含以下权限。

- `ec2:DescribeAddresses`— AWS Ground Station 允许代表您列出所有与 EIPs 之 IPs 关联的内容。
- `ec2:DescribeNetworkInterfaces`— AWS Ground Station 允许代表您获取与 EC2 实例关联的网络接口的信息。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Ground StationAWS 托管策略的更新

查看 AWS Ground Station 自该服务开始跟踪这些更改以来 AWS 托管策略更新的详细信息。要获得有关此页面变更的自动提醒，请订阅 “ AWS Ground Station 文档历史记录” 页面上的 RSS feed。

更改	描述	日期
AWSGroundStationAgentInstancePolicy - 新策略	AWS Ground Station 添加了一项新策略，为数据流终端节点实例提供使用 AWS Ground Station Agent 的权限。	2023 年 4 月 12 日
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy：新策略	AWS Ground Station 添加了一项新策略，该策略 EC2 AWS Ground Station 允许查找与实例关联的公用 IPv4 地址 EIPs 和与 EC2 实例关联的网络接口。	2022 年 11 月 2 日
AWS Ground Station 开始跟踪更改	AWS Ground Station 已开始跟踪 AWS 托管策略的更改。	2021 年 3 月 1 日

在 Ground Station 中使用与服务相关的角色

AWS Ground Station 使用 AWS Identity and Access Management (IAM) 服务相关角色。服务相关角色是一种独特类型的 IAM 角色，它与 Ground Station 直接相关。服务相关角色由 Ground Station 预定义，包括该服务代表您调用其他 AWS 服务所需的所有权限。

服务相关角色可让您更轻松地设置 Ground Station，因为您无需手动添加必要权限。Ground Station 定义其服务相关角色的权限，除非另有定义，否则仅 Ground Station 可以承担该角色。定义的权限包括信任策略和权限策略，而且权限策略不能附加到任何其他 IAM 实体。

有关支持服务相关角色的其他服务的信息，请参阅与 IAM 配合使用的AWS 服务，并在服务相关角色列中查找标有 “是” 的服务。选择是和链接，查看该服务的服务相关角色文档。

Ground Station 的服务相关角色权限

Ground Station 使用名为的服务相关角色 AWSServiceRoleForGroundStationDataflowEndpointGroup— AWS GroundStation 使用此服务相关角色来调用 EC2 以查找公 IPv4 有地址。

AWSServiceRoleForGroundStationDataflowEndpointGroup 服务相关角色信任以下服务来代入该角色：

- `groundstation.amazonaws.com`

名为的角色权限策略 AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy 允许 Ground Station 对指定资源完成以下操作：

- 操作：`all AWS resources (*)` 上的 `ec2:DescribeAddresses`

  Action 允许 Ground Station 列出与之 IPs 关联的所有 EIPs内容。
- 操作：`all AWS resources (*)` 上的 `ec2:DescribeNetworkInterfaces`

  操作允许 Ground Station 获取与 EC2 实例关联的网络接口的相关信息

您必须配置权限，允许 IAM 实体（如用户、组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的服务相关角色权限。

为 Ground Station 创建服务相关角色

您无需手动创建服务相关角色。当你在 AWS CLI 或 AWS API DataflowEndpointGroup 中创建时，Ground Station 会为你创建服务相关角色。

如果您删除该服务相关角色，然后需要再次创建，您可以使用相同流程在账户中重新创建此角色。当您创建时 DataflowEndpointGroup，Ground Station 会再次为您创建服务相关角色。

您还可以使用 IAM 控制台通过向 A mazon 传输数据 EC2用例创建服务相关角色。在 AWS CLI 或 AWS API 中，使用服务名称创建服务相关角色。`groundstation.amazonaws.com`有关更多信息，请参阅 IAM 用户指南 中的创建服务相关角色。如果您删除了此服务相关角色，可以使用同样的过程再次创建角色。

为 Ground Station 编辑服务相关角色

Ground Station 不允许您编辑 AWSServiceRoleForGroundStationDataflowEndpointGroup 服务相关角色。创建服务相关角色后，您将无法更改角色的名称，因为可能有多种实体引用该角色。但是可以使用 IAM 编辑角色描述。有关更多信息，请参阅《IAM 用户指南》中的编辑服务相关角色。

为 Ground Station 删除服务相关角色

如果不再需要使用某个需要服务相关角色的功能或服务，我们建议您删除该角色。这样就没有未被主动监控或维护的未使用实体。

只有在首先 DataflowEndpointGroups 使用服务相关角色删除服务相关角色后，才能删除该角色。这可以保护您免于无意中撤销对您的权限。 DataflowEndpointGroups如果服务相关角色与多个角色一起使用 DataflowEndpointGroups，则必须先删除所有使用 DataflowEndpointGroups 该服务相关角色的角色，然后才能将其删除。

> **Note**
>
> 如果在您尝试删除资源时 Ground Station 服务正在使用该角色，则删除可能会失败。如果发生这种情况，请等待几分钟后重试。

删除 Ground Station 使用的 Ground Station 资源 AWSService RoleForGroundStationDataflowEndpointGroup

- DataflowEndpointGroups 通过 AWS CLI 或 AWS API 删除。

使用 IAM 手动删除服务相关角色

使用 IAM 控制台 AWS CLI、或 AWS API 删除 AWSServiceRoleForGroundStationDataflowEndpointGroup服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的删除服务相关角色。

Ground Station 服务相关角色的受支持区域

Ground Station 支持在服务可用的所有区域使用服务相关角色。有关更多信息，请参阅区域表。

故障排除

NOT_AUTHORIZED_TO_CREATE_SLR-这表示您的账户中用于调用 CreateDataflowEndpointGroup API 的角色没有iam:CreateServiceLinkedRole权限。具有 iam:CreateServiceLinkedRole 权限的管理员须为您的账户手动创建服务相关角色。

静态数据加密 AWS Ground Station

AWS Ground Station 默认提供加密，以使用 AWS 自有的加密密钥保护您的静态敏感数据。

- AWS 拥有的密钥-默认 AWS Ground Station 使用这些密钥自动加密可直接识别的个人数据和星历表。您无法查看、管理或使用 AWS 拥有的密钥，也无法审核其使用情况；但无需采取任何措施或更

改程序来保护加密数据的密钥。有关更多信息，请参阅 AWS Key Management Service 开发人员指南中的 AWS 拥有的密钥。

默认情况下，静态数据加密可帮助降低保护敏感数据时涉及的操作开销和复杂性。同时，它还支持构建符合严格加密合规性以及监管要求的安全应用程序。

AWS Ground Station 对所有敏感的静态数据强制加密，但是，对于某些 AWS Ground Station 资源（例如星历表），您可以选择使用客户托管密钥代替默认的托管密钥。 AWS

- 客户托管密钥-- AWS Ground Station 支持使用您创建、拥有和管理的对称客户托管密钥，在现有 AWS 自有加密的基础上添加第二层加密。由于您可以完全控制这层加密，因此可以执行以下任务：
  - 制定和维护关键策略
  - 建立和维护 IAM 策略和授权
  - 启用和禁用密钥策略
  - 轮换加密材料
  - 添加标签
  - 创建密钥别名
  - 计划要删除的密钥

  有关更多信息，请参阅 AWS Key Management Service 开发人员指南中的客户托管密钥。

下表汇总了 AWS Ground Station 支持使用客户托管密钥的资源

数据类型	AWS 拥有的密钥加密	客户托管密钥加密（可选）
用于计算卫星轨迹的星历数据	已启用	已启用

> **Note**
>
> AWS Ground Station 使用 AWS 自有密钥自动启用静态加密，以免费保护个人身份数据。但是，使用客户托管密钥需支付 AWS KMS 费用。有关定价的更多信息，请参阅 Key Management Service 定价。
> 有关 AWS KMS 的更多信息，请参阅 AWS KMS 开发人员指南。

如何在 AWS KMS 中 AWS Ground Station 使用授权

AWS Ground Station 需要密钥授予才能使用您的客户管理的密钥。

当您上传使用客户托管密钥加密的星历时， AWS Ground Station 会通过向 KMS 发送 CreateGrant 请求来代表您创建密钥授权。 AWS AWS KMS 中的授权用于授予对您账户中的 KMS 密钥的 AWS Ground Station 访问权限。

AWS Ground Station 需要获得授权才能使用您的客户托管密钥进行以下内部操作：

- 向 AWS KMS 发送GenerateDataKey请求以生成由您的客户托管密钥加密的数据密钥。
- 向 AWS KMS 发送解密请求以解密加密的数据密钥，以便它们可用于加密您的数据。
- 向 AWS KMS 发送加密请求以加密提供的数据。

您可以随时撤销授予访问权限，或删除服务对客户托管密钥的访问权限。如果这样做，将 AWS Ground Station 无法访问由客户托管密钥加密的任何数据，这会影响依赖该数据的操作。例如，如果您从当前用于联系人的星历中删除密钥授权，则在接触期间 AWS Ground Station 将无法使用提供的星历数据来指向天线。这将导致该联系以“失败”状态结束。

创建客户托管密钥

您可以使用管理控制台或 KMS 创建对称客户托管 AWS 密钥 APIs。 AWS

创建对称的客户托管式密钥：

按照《密钥管理服务开发人员指南》中创建对称客户托管AWS 密钥的步骤进行操作。

密钥策略

密钥策略控制对客户自主管理型密钥的访问。每个客户托管式密钥必须只有一个密钥策略，其中包含确定谁可以使用密钥以及如何使用密钥的声明。创建客户托管式密钥时，可以指定密钥策略。有关更多信息，请参阅《密钥管理服务开发人员指南》中的管理客户托管密 AWS 钥的访问权限。

要将客户托管密钥 AWS Ground Station 用于您的资源，必须在密钥策略中允许以下 API 操作：

`kms:CreateGrant`：向客户托管密钥添加授权。授予对指定 KMS 密钥的控制访问权限，从而允许对授予操作 AWS Ground Station 所需的访问权限。有关使用授权的更多信息，请参阅 AWS 密钥管理服务开发人员指南。

这 AWS 允许亚马逊执行以下操作：

- 调用 GenerateDataKey 生成加密的数据密钥并将其存储，因为数据密钥不会立即用于加密。
- 调用 Decrypt 使用存储的加密数据密钥访问加密数据。
- 调用 Encryp t 使用数据密钥加密数据。
- 设置停用主体，以允许服务 `RetireGrant`。

`kms:DescribeKey`-提供客户托管的密钥详细信息 AWS Ground Station ，以便在尝试为提供的密钥创建授权之前验证密钥。

以下是您可以为其添加的 IAM 策略声明示例 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" : "*"
  }
]
```

有关在策略中指定权限的更多信息，请参阅 AWS 密钥管理服务开发人员指南。

有关密钥访问疑难解答的更多信息，请参阅 AWS 密钥管理服务开发人员指南。

为指定客户管理的密钥 AWS Ground Station

您可以指定客户托管密钥以加密以下资源：

- 星历

创建资源时，可以通过提供一个来指定数据密钥 kmsKeyArn

- kmsKeyArn- AWS KMS 客户托管密钥的密钥标识符

AWS Ground Station 加密上下文

加密上下文是一组可选的键值对，包含有关数据的其他上下文信息。 AWS KMS 使用加密上下文作为额外的经过身份验证的数据来支持经过身份验证的加密。当您在加密数据的请求中包含加密上下文时，AWS KMS 会将加密上下文绑定到加密数据。要解密数据，您必须在请求中包含相同的加密上下文。

AWS Ground Station 加密上下文

AWS Ground Station 根据要加密的资源使用不同的加密上下文，并为创建的每个密钥授权指定特定的加密上下文。

星历加密上下文：

加密星历资源的密钥授权会绑定到特定的卫星 ARN

```
"encryptionContext": {
```

```
    "aws:groundstation:arn":
 "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
}
```

> **Note**
>
> 密钥授权可重复用于同一密钥卫星。

使用加密上下文进行监控

使用对称的客户托管密钥来加密您的星历表时，您还可以使用审计记录和日志中的加密上下文来识别客户托管密钥的使用情况。加密上下文还会显示在AWS CloudTrail 或 Amazon Logs 生成的 CloudWatch 日志中。

使用加密上下文控制对客户托管式密钥的访问

您可以使用密钥策略和 IAM 策略中的加密上下文作为 `conditions` 来控制对您的对称客户托管密钥的访问。您还可以在授权中使用加密上下文约束。

AWS Ground Station 在授权中使用加密上下文约束来控制对您的账户或区域中客户托管密钥的访问权限。授权约束要求授权允许的操作使用指定的加密上下文。

以下是密钥政策声明示例，用于授予对特定加密上下文的客户托管密钥的访问权限。此策略语句中的条件要求授权具有指定加密上下文的加密上下文约束。

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

监控您的加密密钥 AWS Ground Station

当您在 AWS Ground Station 资源中使用 AWS KMS 客户托管密钥时，您可以使用AWS CloudTrail或Amazon CloudWatch 日志来跟踪 AWS Ground Station 发送到 AWS KMS 的请求。以下示例是CreateGrant、GenerateDataKey、Encrypt和DescribeKey监控 KMS 操作 AWS CloudTrail 的事件Decrypt，这些操作由 G AWS round Station 调用，以访问由您的客户托管密钥加密的数据。

CreateGrant (CloudTrail)

当您使用 AWS KMS 客户托管密钥加密您的星历资源时， AWS Ground Station 会代表您发送访问您账户中的 KMS 密钥的CreateGrant请求。 AWS AWS Ground Station 创建的授权特定于与 AWS KMS 客户托管密钥关联的资源。此外，当您删除资源时，G AWS round Station 会使用该RetireGrant操作来移除授权。

以下示例事件记录 CreateGrant 操作：

```
{
    "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)

当您使用 AWS KMS 客户托管密钥加密您的星历资源时，AWS Ground Station 会代表您发送DescribeKey请求以验证所请求的密钥是否存在于您的账户中。

以下示例事件记录 DescribeKey 操作：

```
{
    "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)

当您使用 AWS KMS 客户托管密钥加密您的星历资源时，AWS Ground Station 会向 KMS 发送GenerateDataKey请求以生成用于加密数据的数据密钥。

以下示例事件记录 GenerateDataKey 操作：

```
{
    "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)

当您使用 AWS KMS 客户托管密钥加密您的星历资源时，如果提供的星历已 AWS Ground Station 使用相同的客户托管密钥加密，则使用该Decrypt操作来解密所提供的星历表。例如，如果从 S3 存储桶上传星历并使用给定密钥对该桶中星历进行加密。

以下示例事件记录 Decrypt 操作：

```
{
    "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"
}
```

传输期间的数据加密 AWS Ground Station

AWS Ground Station 默认提供加密功能，以便在传输过程中保护您的敏感数据。根据任务配置文件配置，可以通过两种方式在 AWS Ground Station 天线位置和您的 Amazon EC2 实例之间传输数据。

- AWS Ground Station 代理人
- 数据流端点

每种流式传输数据方法对传输中数据的加密处理方式都不一样。下文将介绍每种方法。

AWS Ground Station 代理直播

AWS Ground Station 代理使用客户管理的 AWS KMS 密钥对其直播进行加密。在您的 Amazon EC2 实例上运行的 AWS Ground Station 代理将自动解密流以提供解密后的数据。

用于加密直播的 AWS KMS 密钥是在 streamsKmsKey参数`MissionProfile`中创建时指定的。所有授予密钥 AWS Ground Station 访问权限的权限均通过所附的 AWS KMS 密钥策略进行处理`streamsKmsKey`。

数据流端点流

数据流端点流使用数据报传输层安全性 (DTLS) 进行加密。这是使用自签名证书完成的，不需要额外的配置。

任务配置文件配置示例

所提供的示例说明了如何使用公共广播卫星并创建支持该卫星的任务概况。生成的模板旨在帮助您与公共广播卫星进行联系，并帮助您做出有关卫星的决定。

主题

JPSS-1-公共广播卫星 (PBS)-评估

此示例部分与客户入职流程概述. 它提供了与 AWS Ground Station 以下具体示例的简短兼容性分析，并为其奠定了基础。

如公共广播卫星本节所述，您可以使用公开可用的特定卫星或卫星的通信路径。在本节中，我们用 AWS Ground Station 术语描述 JPSS-1。作为参考，我们使用联合极地卫星系统 1 (JPSS-1) 航天器高速率数据 (HRD) 直接广播电台 (DBS) 射频 (RF) 接口控制文档 (ICD) 来完成示例。另外，值得注意的是，JPSS-1 与 NORAD ID 43013 有关。

JPSS-1 卫星提供一条上行链路和三条直接下行链路通信路径，如 ICD 图 1-1 所示。在这四条通信路径中，只有一条高速率数据 (HRD) 下行链路通信路径可供公众使用。基于此，你会看到这条路径还将有更具体的数据与之相关联。四条路径如下所示：

- 命令路径（上行链路），MHz 中心频率为 2067.27，数据速率为 2-128 kbps。此路径不可公开访问。
- MHz 中心频率为 2247.5 的遥测路径（下行链路），数据速率为 1-524 kbps。此路径不可公开访问。
- GHz 中心频率为 26.7034 的 SMD 路径（下行链路），数据速率为 150-300 Mbps。此路径不可公开访问。
- HRD 路径（下行链路）的 RF，MHz 中心频率为 7812，数据速率为 15 Mbps。它的 MHz 带宽为 30，而且确实如此 right-hand-circular-polarized。当您搭载 JPSS-1 时 AWS Ground Station，这是

您可以访问的通信路径。该通信路径包含仪器科学数据、仪器工程数据、仪器遥测数据和实时航天器内部管理数据。

当我们比较潜在的数据路径时，我们发现命令（上行链路）、遥测（下行链路）和 HRD（下行链路）路径符合的频率、带宽和多通道并发使用能力。 AWS Ground Station SMD 路径不兼容，因为中心频率超出了现有接收器的范围。有关支持的功能的更多信息，请参阅AWS Ground Station 网站能力。

> **Note**
>
> 由于 SMD 路径与之不兼容 AWS Ground Station ，因此示例配置中不会显示。

> **Note**
>
> 由于命令（上行链路）和遥测（下行链路）路径未在 ICD 中定义，也不可供公众使用，因此使用时提供的值是名义值。

使用 Amazon S3 数据传输的公共广播卫星

此示例建立在用户指南JPSS-1-公共广播卫星 (PBS)-评估 部分所做的分析的基础上。

在本示例中，你需要假设一个场景，即你想将 HRD 通信路径捕获为数字中频，并将其存储起来以备将来的批处理之用。这样可以节省数字化后的原始射频 (RF) 同相正交 (I/Q) 样本。将数据放入 Amazon S3 存储桶后，您可以使用所需的任何软件对数据进行解调和解码。有关处理的详细示例，请参阅 MathWorks 教程。使用此示例后，您可以考虑添加 Amazon EC2 现货定价组件来处理数据并降低总体处理成本。

通信路径

本节介绍规划您的数据流通信路径入门。

以下所有模板片段都属于 AWS CloudFormation 模板的 “资源” 部分。

```
Resources:
  # Resources that you would like to create should be placed within the Resources
 section.
```

> **Note**
>
> 有关 AWS CloudFormation 模板内容的更多信息，请参阅模板部分。

考虑到我们向 Amazon S3 提供单一通信路径的场景，您知道您将拥有一条异步传输路径。根据本异步数据传输节，您必须定义一个 Amazon S3 存储桶。

```
  # The S3 bucket where AWS Ground Station will deliver the downlinked data.
  GroundStationS3DataDeliveryBucket:
    Type: AWS::S3::Bucket
    DeletionPolicy: Retain
    UpdateReplacePolicy: Retain
    Properties:
      # Results in a bucket name formatted like: aws-groundstation-data-{account id}-
{region}-{random 8 character string}
      BucketName: !Join ["-", ["aws-groundstation-data", !Ref AWS::AccountId, !Ref
 AWS::Region, !Select [0, !Split ["-", !Select [2, !Split ["/", !Ref AWS::StackId]]]]]]
```

此外，您还需要创建相应的角色和策略 AWS Ground Station 才能允许使用存储桶。

```
  # The IAM role that AWS Ground Station will assume to have permission find and write
  # data to your S3 bucket.
  GroundStationS3DataDeliveryRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Statement:
          - Action:
              - 'sts:AssumeRole'
            Effect: Allow
            Principal:
              Service:
                - groundstation.amazonaws.com
            Condition:
              StringEquals:
                "aws:SourceAccount": !Ref AWS::AccountId
```

```
              ArnLike:
                "aws:SourceArn": !Sub "arn:aws:groundstation:${AWS::Region}:
${AWS::AccountId}:config/s3-recording/*"

  # The S3 bucket policy that defines what actions AWS Ground Station can perform on
 your S3 bucket.
  GroundStationS3DataDeliveryBucketPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Statement:
          - Action:
              - 's3:GetBucketLocation'
            Effect: Allow
            Resource:
              - !GetAtt GroundStationS3DataDeliveryBucket.Arn
          - Action:
              - 's3:PutObject'
            Effect: Allow
            Resource:
              - !Join [ "/", [ !GetAtt GroundStationS3DataDeliveryBucket.Arn, "*" ] ]
      PolicyName: GroundStationS3DataDeliveryPolicy
      Roles:
        - !Ref GroundStationS3DataDeliveryRole
```

AWS Ground Station 配置

本节介绍创建配置入门。

你需要一个跟踪配置来设置你使用自动追踪的偏好。选择 P REFERRED 作为自动跟踪可以提高信号质量，但由于 JPSS-1 星历质量足够，因此不需要满足信号质量。

```
  TrackingConfig:
    Type: AWS::GroundStation::Config
    Properties:
      Name: "JPSS Tracking Config"
      ConfigData:
        TrackingConfig:
          Autotrack: "PREFERRED"
```

根据通信路径，您需要定义一个天线下行链路配置来表示卫星部分，并定义一个 s3 录音以引用您刚刚创建的 Amazon S3 存储桶。

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 JpssDownlinkDigIfAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Downlink DigIF Antenna Config"
     ConfigData:
       AntennaDownlinkConfig:
         SpectrumConfig:
           Bandwidth:
             Units: "MHz"
             Value: 30
           CenterFrequency:
             Units: "MHz"
             Value: 7812
           Polarization: "RIGHT_HAND"

 # The AWS Ground Station S3 Recording Config that defines the S3 bucket and IAM role
to use
 # when AWS Ground Station delivers the downlink data.
 S3RecordingConfig:
   Type: AWS::GroundStation::Config
   DependsOn: GroundStationS3DataDeliveryBucketPolicy
   Properties:
     Name: "JPSS S3 Recording Config"
     ConfigData:
       S3RecordingConfig:
         BucketArn: !GetAtt GroundStationS3DataDeliveryBucket.Arn
         RoleArn: !GetAtt GroundStationS3DataDeliveryRole.Arn
```

AWS Ground Station 任务简介

本节介绍创建任务档案入门。

现在你已经有了相关的配置，你可以用它们来构造数据流。其余参数将使用默认值。

```
  # The AWS Ground Station Mission Profile that groups the above configurations to
define how to downlink data.
 JpssAsynchMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: "43013 JPSS Asynchronous Data"
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Ref JpssDownlinkDigIfAntennaConfig
         Destination: !Ref S3RecordingConfig
```

把它放在一起

利用上述资源，您现在可以安排 JPSS-1 联系人从任何已上线人员进行异步数据传输。 AWS Ground Station AWS Ground Station 地点

以下是一个完整的 AWS CloudFormation 模板，其中包括本节中描述的所有资源，这些资源组合成一个可以直接在中使用的模板 AWS CloudFormation。

名为的 AWS CloudFormation 模板`AquaSnppJpss-1TerraDigIfS3DataDelivery.yml`包含一个 Amazon S3 存储桶以及安排联系和接收 VITA-49 信号/IP 直接广播数据所需的 AWS Ground Station 资源。

如果你的账户未登录 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra，请参阅。机载卫星

> **Note**
>
> 您可以使用有效 AWS 凭证访问客户登录 Amazon S3 存储桶，从而访问模板。以下链接使用区域性 Amazon S3 存储桶。更改`us-west-2`区域代码以表示要在其中创建 AWS CloudFormation 堆栈的相应区域。
> 此外，以下说明使用 YAML。但是，模板有 YAML 和 JSON 这两种格式。要使用 JSON，请在下载模板`.json`时将`.yml`文件扩展名替换为。

要使用下载模板 AWS CLI，请使用以下命令：

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

在浏览器中导航到以下 URL，可以在控制台中查看和下载此模板：

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

您可以使用以下链接直接在中 AWS CloudFormation 指定模板：

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

利用数据流端点（窄带）的公共广播卫星

此示例建立在用户指南JPSS-1-公共广播卫星 (PBS)-评估 部分所做的分析的基础上。

要完成此示例，您需要假设一个场景，即您要将 HRD 通信路径捕获为数字中频 (digiF)，并在使用 SDR 的 A EC2 mazon 实例上的数据流终端节点应用程序接收到时对其进行处理。

通信路径

本节介绍规划您的数据流通信路径入门。在本示例中，您将在 AWS CloudFormation 模板中创建两个部分：“参数” 和 “资源” 部分。

> **Note**
>
> 有关 AWS CloudFormation 模板内容的更多信息，请参阅模板部分。

在 “参数” 部分，您将添加以下参数。在通过 AWS CloudFormation 控制台创建堆栈时，您将为这些值指定值。

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  ReceiverAMI:
```

```
    Description: The Ground Station DDX AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> 您需要创建密钥对，并提供 Amazon EC2 EC2Key 参数的名称。请参阅为您的 Amazon EC2 实例创建密钥对。
>
> 此外，在创建 AWS CloudFormation 堆栈时，您需要提供正确的特定于区域的 AMI ID。请参阅AWS Ground Station Amazon 机器映像 (AMIs)。

其余的模板片段属于 AWS CloudFormation 模板的 “资源” 部分。

```
Resources:
  # Resources that you would like to create should be placed within the resource
 section.
```

考虑到我们为 EC2 实例提供单一通信路径的场景，您将拥有一条同步传输路径。根据本同步数据传输节，您必须使用数据流终端节点应用程序设置和配置一个 Amazon EC2 实例，并创建一个或多个数据流终端节点组。

```
  # The EC2 instance that will send/receive data to/from your satellite using AWS
 Ground Station.
  ReceiverInstance:
    Type: AWS::EC2::Instance
    Properties:
      DisableApiTermination: false
      IamInstanceProfile: !Ref GeneralInstanceProfile
      ImageId: !Ref ReceiverAMI
      InstanceType: m5.4xlarge
      KeyName: !Ref EC2Key
      Monitoring: true
      PlacementGroupName: !Ref ClusterPlacementGroup
      SecurityGroupIds:
        - Ref: InstanceSecurityGroup
      SubnetId: !Ref ReceiverSubnet
```

```
      BlockDeviceMappings:
        - DeviceName: /dev/xvda
          Ebs:
            VolumeType: gp2
            VolumeSize: 40
      Tags:
        - Key: Name
          Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
      UserData:
        Fn::Base64:
          |
          #!/bin/bash
          exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
2>&1
          echo `date +'%F %R:%S'` "INFO: Logging Setup" >&2

          GROUND_STATION_DIR="/opt/aws/groundstation"
          GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
          STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"

          echo "Creating ${STREAM_CONFIG_PATH}"
          cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
          {
            "ddx_streams": [
              {
                "streamName": "Downlink",
                "maximumWanRate": 4000000000,
                "lanConfigDevice": "lo",
                "lanConfigPort": 50000,
                "wanConfigDevice": "eth1",
                "wanConfigPort": 55888,
                "isUplink": false
              }
            ]
          }
          STREAM_CONFIG

          echo "Waiting for dataflow endpoint application to start"
          while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

          echo "Configuring dataflow endpoint application streams"
          python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
          sleep 2
```

```
          python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

          exit 0

 # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
 # Station will use to send/receive data to/from your satellite.
 DataflowEndpointGroup:
   Type: AWS::GroundStation::DataflowEndpointGroup
   Properties:
     ContactPostPassDurationSeconds: 180
     ContactPrePassDurationSeconds: 120
     EndpointDetails:
       - Endpoint:
           Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
           Address:
             Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
             Port: 55888
         SecurityDetails:
           SecurityGroupIds:
             - Ref: "DataflowEndpointSecurityGroup"
           SubnetIds:
             - !Ref ReceiverSubnet
           RoleArn: !GetAtt DataDeliveryServiceRole.Arn

 # The security group for your EC2 instance.
 InstanceSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: AWS Ground Station receiver instance security group.
     VpcId: !Ref ReceiverVPC
     SecurityGroupIngress:
       # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
         Description: "AWS Ground Station Downlink Stream"

 # The security group that the ENI created by AWS Ground Station belongs to.
 DataflowEndpointSecurityGroup:
   Type: AWS::EC2::SecurityGroup
```

```
    Properties:
      GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
      VpcId: !Ref ReceiverVPC
      SecurityGroupEgress:
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 10.0.0.0/8
          Description: "AWS Ground Station Downlink Stream To 10/8"
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 172.16.0.0/12
          Description: "AWS Ground Station Downlink Stream To 172.16/12"
        - IpProtocol: udp
          FromPort: 55888
          ToPort: 55888
          CidrIp: 192.168.0.0/16
          Description: "AWS Ground Station Downlink Stream To 192.168/16"

  # The placement group in which your EC2 instance is placed.
  ClusterPlacementGroup:
    Type: AWS::EC2::PlacementGroup
    Properties:
      Strategy: cluster

  ReceiverVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: "10.0.0.0/16"
      Tags:
        - Key: "Name"
          Value: "AWS Ground Station - PBS to dataflow endpoint Example VPC"
        - Key: "Description"
          Value: "VPC for EC2 instance receiving AWS Ground Station data"

  ReceiverSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      # Ensure your CidrBlock will always have at least one available IP address per
dataflow endpoint.
      # See https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html for
subent sizing guidelines.
```

```
      CidrBlock: "10.0.0.0/24"
      Tags:
        - Key: "Name"
          Value: "AWS Ground Station - PBS to dataflow endpoint Example Subnet"
        - Key: "Description"
          Value: "Subnet for EC2 instance receiving AWS Ground Station data"
      VpcId: !Ref ReceiverVPC

  # An ENI providing a fixed IP address for AWS Ground Station to connect to.
  ReceiverInstanceNetworkInterface:
    Type: AWS::EC2::NetworkInterface
    Properties:
      Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
      GroupSet:
        - !Ref InstanceSecurityGroup
      SubnetId: !Ref ReceiverSubnet

  # Attach the ENI to the EC2 instance.
  ReceiverInstanceInterfaceAttachment:
    Type: AWS::EC2::NetworkInterfaceAttachment
    Properties:
      DeleteOnTermination: false
      DeviceIndex: "1"
      InstanceId: !Ref ReceiverInstance
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface
```

此外，您还需要创建适当的策略和角色， AWS Ground Station 以允许在您的账户中创建弹性网络接口 (ENI)。

```
  # AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
  DataDeliveryServiceRole:
    Type: AWS::IAM::Role
    Properties:
      Policies:
        - PolicyDocument:
            Statement:
              - Action:
                  - ec2:CreateNetworkInterface
                  - ec2:DeleteNetworkInterface
```

```
                - ec2:CreateNetworkInterfacePermission
                - ec2:DeleteNetworkInterfacePermission
                - ec2:DescribeSubnets
                - ec2:DescribeVpcs
                - ec2:DescribeSecurityGroups
              Effect: Allow
              Resource: '*'
          Version: '2012-10-17'
        PolicyName: DataDeliveryServicePolicy
    AssumeRolePolicyDocument:
      Version: 2012-10-17
      Statement:
        - Effect: Allow
          Principal:
            Service:
            - groundstation.amazonaws.com
          Action:
          - sts:AssumeRole

# The EC2 instance assumes this role.
InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Service:
              - "ec2.amazonaws.com"
          Action:
            - "sts:AssumeRole"
    Path: "/"
    ManagedPolicyArns:
      - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
      - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
      - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
      - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM

# The instance profile for your EC2 instance.
GeneralInstanceProfile:
  Type: AWS::IAM::InstanceProfile
  Properties:
    Roles:
```

```
        - !Ref InstanceRole
```

AWS Ground Station 配置

本节介绍创建配置入门。

你需要一个跟踪配置来设置你使用自动追踪的偏好。选择 P REFERRED 作为自动跟踪可以提高信号质量，但由于 JPSS-1 星历质量足够，因此不需要满足信号质量。

```
 TrackingConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Tracking Config"
     ConfigData:
       TrackingConfig:
         Autotrack: "PREFERRED"
```

根据通信路径，你需要定义一个天线下行链路配置来表示卫星部分，以及一个数据流端点配置来引用定义端点详细信息的数据流端点组。

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 SnppJpssDownlinkDigIfAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "SNPP JPSS Downlink DigIF Antenna Config"
     ConfigData:
       AntennaDownlinkConfig:
         SpectrumConfig:
           Bandwidth:
             Units: "MHz"
             Value: 30
           CenterFrequency:
             Units: "MHz"
             Value: 7812
           Polarization: "RIGHT_HAND"
```

```
 # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
 # from your satellite.
 DownlinkDigIfEndpointConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "Aqua SNPP JPSS Downlink DigIF Endpoint Config"
     ConfigData:
       DataflowEndpointConfig:
         DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
         DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station 任务简介

本节介绍创建任务档案入门。

现在你已经有了相关的配置，你可以用它们来构造数据流。其余参数将使用默认值。

```
 # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
 # uplink and downlink data to your satellite.
 SnppJpssMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: "37849 SNPP And 43013 JPSS"
     ContactPrePassDurationSeconds: 120
     ContactPostPassDurationSeconds: 60
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Ref SnppJpssDownlinkDigIfAntennaConfig
         Destination: !Ref DownlinkDigIfEndpointConfig
```

把它放在一起

利用上述资源，您现在可以安排 JPSS-1 联系人，以便从任何已上线人员同步传送数据。 AWS Ground Station AWS Ground Station 地点

以下是一个完整的 AWS CloudFormation 模板，其中包括本节中描述的所有资源，这些资源组合成一个可以直接在中使用的模板 AWS CloudFormation。

名AquaSnppJpssTerraDigIF.yml为的 AWS CloudFormation 模板旨在让你快速访问开始接收 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 卫星的数字化中频 (digiF) 数据。它包含一个 Amazon EC2 实例和接收原始 digiF 直接广播数据所需的 AWS CloudFormation 资源。

如果你的账户未登录 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra，请参阅。机载卫星

> **Note**
>
> 您可以使用有效 AWS 凭证访问客户登录 Amazon S3 存储桶，从而访问该模板。以下链接使用区域性 Amazon S3 存储桶。更改us-west-2区域代码以表示要在其中创建 AWS CloudFormation 堆栈的相应区域。
> 此外，以下说明使用 YAML。但是，模板有 YAML 和 JSON 这两种格式。要使用 JSON，请在下载模板.json时将.yml文件扩展名替换为。

要使用下载模板 AWS CLI，请使用以下命令：

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

在浏览器中导航到以下 URL，可以在控制台中查看和下载此模板：

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

您可以使用以下链接直接在中 AWS CloudFormation 指定模板：

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

该模板还定义了哪些其他资源？

该AquaSnppJpssTerraDigIF模板包括以下其他资源：

- （可选）CloudWatch 事件触发器-使用联系 AWS Ground Station 前后发送 CloudWatch 的事件触发的 AWS Lambda 函数。该 AWS Lambda 函数将启动并有选择地停止您的接收器实例。

- （可选）联系人EC2 验证-使用 Lambda 为带有 SNS 通知的联系人设置亚马逊 EC2 实例的验证系统的选项。需要注意的是，这可能会产生费用，具体取决于您当前的使用情况。
- Ground Station 亚马逊机器映像检索 Lambda：用于选择您的实例中安装的软件以及您选择的 AMI 的选项。软件选项包括 `DDX 2.6.2 Only` 和 `DDX 2.6.2 with qRadio 3.6.0`。随着更多软件更新和功能的发布，这些选项将继续扩展。
- 其他任务概况 ——其他公共广播卫星（Aqua、SNPP 和 Terra）的任务概况。
- 其他天线下行链路配置——其他公共广播卫星（Aqua、SN PP 和 Terra）的天线下行链路配置。

已填充此模板中卫星的值和参数。这些参数使您可以轻松地 AWS Ground Station 立即使用这些卫星。使用此模板 AWS Ground Station 时，您无需配置自己的值即可使用。但是，您可以自定义这些值以使模板适用于您的使用案例。

我可以在哪里接收我的数据？

数据流终端节点组设置为使用此模板的一部分创建的接收实例网络接口。接收器实例使用数据流端点应用程序从数据流端点定义 AWS Ground Station 的端口接收数据流。接收到数据后，可通过接收实例的环回适配器上的 UDP 端口 50000 使用数据。有关设置数据流终端节点组的更多信息，请参阅。AWS::GroundStation::DataflowEndpointGroup

使用数据流端点的公共广播卫星（解调和解码）

此示例建立在用户指南JPSS-1-公共广播卫星 (PBS)-评估 部分所做的分析的基础上。

要完成此示例，您需要假设一个场景，即要使用数据流端点将 HRD 通信路径捕获为解调和解码的直接广播数据。如果您计划使用美国宇航局直接读取实验室软件（RT-STPS和IPOPP）处理数据，则此示例是一个不错的起点。

通信路径

本节介绍规划您的数据流通信路径入门。在本示例中，您将在 AWS CloudFormation 模板中创建两个部分：“参数” 和 “资源” 部分。

> **Note**
>
> 有关 AWS CloudFormation 模板内容的更多信息，请参阅模板部分。

在 “参数” 部分，您将添加以下参数。在通过 AWS CloudFormation 控制台创建堆栈时，您将为这些值指定值。

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  ReceiverAMI:
    Description: The Ground Station DDX AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> 您需要创建密钥对，并提供 Amazon EC2 EC2Key 参数的名称。请参阅为您的 Amazon EC2 实例创建密钥对。
>
> 此外，在创建 AWS CloudFormation 堆栈时，您需要提供正确的特定于区域的 AMI ID。请参阅 AWS Ground Station Amazon 机器映像 (AMIs)。

其余模板片段属于 AWS CloudFormation 模板的 “资源” 部分。

```
Resources:
  # Resources that you would like to create should be placed within the resource
 section.
```

考虑到我们为 EC2 实例提供单一通信路径的场景，您将拥有一条同步传输路径。根据本同步数据传输节，您必须使用数据流终端节点应用程序设置和配置一个 Amazon EC2 实例，并创建一个或多个数据流终端节点组。

```
 # The EC2 instance that will send/receive data to/from your satellite using AWS
Ground Station.
 ReceiverInstance:
   Type: AWS::EC2::Instance
   Properties:
     DisableApiTermination: false
     IamInstanceProfile: !Ref GeneralInstanceProfile
     ImageId: !Ref ReceiverAMI
     InstanceType: m5.4xlarge
     KeyName: !Ref EC2Key
     Monitoring: true
     PlacementGroupName: !Ref ClusterPlacementGroup
     SecurityGroupIds:
       - Ref: InstanceSecurityGroup
     SubnetId: !Ref ReceiverSubnet
     BlockDeviceMappings:
       - DeviceName: /dev/xvda
         Ebs:
           VolumeType: gp2
           VolumeSize: 40
     Tags:
       - Key: Name
         Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
     UserData:
       Fn::Base64:
         |
         #!/bin/bash
         exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
2>&1
         echo `date +'%F %R:%S'` "INFO: Logging Setup" >&2

         GROUND_STATION_DIR="/opt/aws/groundstation"
         GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
         STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"

         echo "Creating ${STREAM_CONFIG_PATH}"
         cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
         {
           "ddx_streams": [
             {
               "streamName": "Downlink",
               "maximumWanRate": 4000000000,
               "lanConfigDevice": "lo",
               "lanConfigPort": 50000,
```

```
                "wanConfigDevice": "eth1",
                "wanConfigPort": 55888,
                "isUplink": false
              }
            ]
          }
          STREAM_CONFIG

          echo "Waiting for dataflow endpoint application to start"
          while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

          echo "Configuring dataflow endpoint application streams"
          python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
          sleep 2
          python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

          exit 0
```

```
 # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
 # Station will use to send/receive data to/from your satellite.
 DataflowEndpointGroup:
   Type: AWS::GroundStation::DataflowEndpointGroup
   Properties:
     ContactPostPassDurationSeconds: 180
     ContactPrePassDurationSeconds: 120
     EndpointDetails:
       - Endpoint:
           Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
           Address:
             Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
             Port: 55888
         SecurityDetails:
           SecurityGroupIds:
             - Ref: "DataflowEndpointSecurityGroup"
           SubnetIds:
             - !Ref ReceiverSubnet
           RoleArn: !GetAtt DataDeliveryServiceRole.Arn

 # The security group that the ENI created by AWS Ground Station belongs to.
```

```
 DataflowEndpointSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
     VpcId: !Ref ReceiverVPC
     SecurityGroupEgress:
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 10.0.0.0/8
         Description: "AWS Ground Station Downlink Stream To 10/8"
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 172.16.0.0/12
         Description: "AWS Ground Station Downlink Stream To 172.16/12"
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         CidrIp: 192.168.0.0/16
         Description: "AWS Ground Station Downlink Stream To 192.168/16"

 # The placement group in which your EC2 instance is placed.
 ClusterPlacementGroup:
   Type: AWS::EC2::PlacementGroup
   Properties:
     Strategy: cluster

 # The security group for your EC2 instance.
 InstanceSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: AWS Ground Station receiver instance security group.
     VpcId: !Ref ReceiverVPC
     SecurityGroupIngress:
       # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
       - IpProtocol: udp
         FromPort: 55888
         ToPort: 55888
         SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
         Description: "AWS Ground Station Downlink Stream"
```

```
 ReceiverVPC:
   Type: AWS::EC2::VPC
   Properties:
     CidrBlock: "10.0.0.0/16"
     Tags:
       - Key: "Name"
         Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
VPC"
       - Key: "Description"
         Value: "VPC for EC2 instance receiving AWS Ground Station data"

 ReceiverSubnet:
   Type: AWS::EC2::Subnet
   Properties:
     CidrBlock: "10.0.0.0/24"
     Tags:
       - Key: "Name"
         Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
Subnet"
       - Key: "Description"
         Value: "Subnet for EC2 instance receiving AWS Ground Station data"
     VpcId: !Ref ReceiverVPC

 # An ENI providing a fixed IP address for AWS Ground Station to connect to.
 ReceiverInstanceNetworkInterface:
   Type: AWS::EC2::NetworkInterface
   Properties:
     Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
     GroupSet:
       - !Ref InstanceSecurityGroup
     SubnetId: !Ref ReceiverSubnet

 # Attach the ENI to the EC2 instance.
 ReceiverInstanceInterfaceAttachment:
   Type: AWS::EC2::NetworkInterfaceAttachment
   Properties:
     DeleteOnTermination: false
     DeviceIndex: "1"
     InstanceId: !Ref ReceiverInstance
     NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

 # The instance profile for your EC2 instance.
 GeneralInstanceProfile:
```

```
    Type: AWS::IAM::InstanceProfile
    Properties:
      Roles:
        - !Ref InstanceRole
```

您还需要相应的策略、角色和配置文件，AWS Ground Station 以便在您的账户中创建弹性网络接口 (ENI)。

```
 # AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
 DataDeliveryServiceRole:
   Type: AWS::IAM::Role
   Properties:
     Policies:
       - PolicyDocument:
           Statement:
             - Action:
                 - ec2:CreateNetworkInterface
                 - ec2:DeleteNetworkInterface
                 - ec2:CreateNetworkInterfacePermission
                 - ec2:DeleteNetworkInterfacePermission
                 - ec2:DescribeSubnets
                 - ec2:DescribeVpcs
                 - ec2:DescribeSecurityGroups
               Effect: Allow
               Resource: '*'
           Version: '2012-10-17'
         PolicyName: DataDeliveryServicePolicy
     AssumeRolePolicyDocument:
       Version: 2012-10-17
       Statement:
         - Effect: Allow
           Principal:
             Service:
             - groundstation.amazonaws.com
           Action:
           - sts:AssumeRole

 # The EC2 instance assumes this role.
 InstanceRole:
   Type: AWS::IAM::Role
```

```
    Properties:
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Principal:
              Service:
                - "ec2.amazonaws.com"
            Action:
              - "sts:AssumeRole"
      Path: "/"
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
        - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
        - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
        - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM
```

AWS Ground Station 配置

本节创建配置代表用户指南。

你需要一个跟踪配置来设置你使用自动追踪的偏好。选择 P REFERRED 作为自动跟踪可以提高信号质量，但由于 JPSS-1 星历质量足够，因此不需要满足信号质量。

```
TrackingConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Tracking Config"
    ConfigData:
      TrackingConfig:
        Autotrack: "PREFERRED"
```

根据通信路径，您需要定义一个antenna-downlink-demod-decode配置来表示卫星部分，以及一个数据流端点配置来引用定义端点详细信息的数据流端点组。

> **Note**
>
> 有关如何为和设置值的DemodulationConfig详细信息DecodeConfig，请参阅天线下行传输解调解码配置。

```
 # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
 # downlink data from your satellite.
 JpssDownlinkDemodDecodeAntennaConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "JPSS Downlink Demod Decode Antenna Config"
     ConfigData:
       AntennaDownlinkDemodDecodeConfig:
         SpectrumConfig:
           CenterFrequency:
             Value: 7812
             Units: "MHz"
           Polarization: "RIGHT_HAND"
           Bandwidth:
             Value: 30
             Units: "MHz"
         DemodulationConfig:
           UnvalidatedJSON: '{
             "type":"QPSK",
             "qpsk":{
               "carrierFrequencyRecovery":{
                 "centerFrequency":{
                   "value":7812,
                   "units":"MHz"
                 },
                 "range":{
                   "value":250,
                   "units":"kHz"
                 }
               },
               "symbolTimingRecovery":{
                 "symbolRate":{
                   "value":15,
                   "units":"Msps"
```

```
                },
                "range":{
                  "value":0.75,
                  "units":"ksps"
                },
                "matchedFilter":{
                  "type":"ROOT_RAISED_COSINE",
                  "rolloffFactor":0.5
                }
              }
            }
          }'
        DecodeConfig:
          UnvalidatedJSON: '{
            "edges":[
              {
                "from":"I-Ingress",
                "to":"IQ-Recombiner"
              },
              {
                "from":"Q-Ingress",
                "to":"IQ-Recombiner"
              },
              {
                "from":"IQ-Recombiner",
                "to":"CcsdsViterbiDecoder"
              },
              {
                "from":"CcsdsViterbiDecoder",
                "to":"NrzmDecoder"
              },
              {
                "from":"NrzmDecoder",
                "to":"UncodedFramesEgress"
              }
            ],
            "nodeConfigs":{
              "I-Ingress":{
                "type":"CODED_SYMBOLS_INGRESS",
                "codedSymbolsIngress":{
                  "source":"I"
                }
              },
              "Q-Ingress":{
```

```
                "type":"CODED_SYMBOLS_INGRESS",
                "codedSymbolsIngress":{
                  "source":"Q"
                }
              },
              "IQ-Recombiner":{
                "type":"IQ_RECOMBINER"
              },
              "CcsdsViterbiDecoder":{
                "type":"CCSDS_171_133_VITERBI_DECODER",
                "ccsds171133ViterbiDecoder":{
                  "codeRate":"ONE_HALF"
                }
              },
              "NrzmDecoder":{
                "type":"NRZ_M_DECODER"
              },
              "UncodedFramesEgress":{
                "type":"UNCODED_FRAMES_EGRESS"
              }
            }
          }'
```

```
 # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
 # from your satellite.
 DownlinkDemodDecodeEndpointConfig:
   Type: AWS::GroundStation::Config
   Properties:
     Name: "Aqua SNPP JPSS Downlink Demod Decode Endpoint Config"
     ConfigData:
       DataflowEndpointConfig:
         DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
         DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station 任务简介

本节创建任务档案代表用户指南。

现在你已经有了相关的配置，你可以用它们来构造数据流。其余参数将使用默认值。

```
 # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
 # uplink and downlink data to your satellite.
 SnppJpssMissionProfile:
   Type: AWS::GroundStation::MissionProfile
   Properties:
     Name: "37849 SNPP And 43013 JPSS"
     ContactPrePassDurationSeconds: 120
     ContactPostPassDurationSeconds: 60
     MinimumViableContactDurationSeconds: 180
     TrackingConfigArn: !Ref TrackingConfig
     DataflowEdges:
       - Source: !Join [ "/", [ !Ref JpssDownlinkDemodDecodeAntennaConfig,
"UncodedFramesEgress" ] ]
         Destination: !Ref DownlinkDemodDecodeEndpointConfig
```

把它放在一起

利用上述资源，您现在可以安排 JPSS-1 联系人，以便从任何已上线人员同步传送数据。 AWS Ground Station AWS Ground Station 地点

以下是一个完整的 AWS CloudFormation 模板，其中包括本节中描述的所有资源，这些资源组合成一个可以直接在中使用的模板 AWS CloudFormation。

名AquaSnppJpss.yml为的 AWS CloudFormation 模板旨在让你快速访问开始接收 Aqua、SNPP 和 JPSS-1/NOAA-20 卫星的数据。它包含一个 Amazon EC2 实例和安排联系以及接收解调和解码后的直接广播数据所需的 AWS Ground Station 资源。

如果你的账户未登录 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra，请参阅。机载卫星

> **Note**
>
> 您可以使用有效 AWS 凭证访问客户登录 Amazon S3 存储桶，从而访问模板。以下链接使用区域性 Amazon S3 存储桶。更改us-west-2区域代码以表示要在其中创建 AWS CloudFormation 堆栈的相应区域。
> 此外，以下说明使用 YAML。但是，模板有 YAML 和 JSON 这两种格式。要使用 JSON，请在下载模板.json时将.yml文件扩展名替换为。

要使用下载模板 AWS CLI，请使用以下命令：

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

在浏览器中导航到以下 URL，可以在控制台中查看和下载此模板：

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

您可以使用以下链接直接在中 AWS CloudFormation 指定模板：

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

该模板定义了哪些其他资源？

该AquaSnppJpss模板包括以下其他资源：

- （可选）CloudWatch 事件触发器-使用联系 AWS Ground Station 前后发送 CloudWatch 的事件触发的 AWS Lambda 函数。该 AWS Lambda 函数将启动并有选择地停止您的接收器实例。
- （可选）联系人EC2 验证-使用 Lambda 为带有 SNS 通知的联系人设置亚马逊 EC2 实例的验证系统的选项。需要注意的是，这可能会产生费用，具体取决于您当前的使用情况。
- Ground Station 亚马逊机器映像检索 Lambda：用于选择您的实例中安装的软件以及您选择的 AMI 的选项。软件选项包括 `DDX 2.6.2 Only` 和 `DDX 2.6.2 with qRadio 3.6.0`。如果要使用宽带 digiF 数据传输和代理 AWS Ground Station ，请参阅。使用 AWS Ground Station 代理（宽带）的公共广播卫星随着更多软件更新和功能的发布，这些选项将继续扩展。
- 其他任务概况 ——其他公共广播卫星（Aqua、SNPP 和 Terra）的任务概况。
- 其他天线下行链路配置——其他公共广播卫星（Aqua、SN PP 和 Terra）的天线下行链路配置。

已填充此模板中卫星的值和参数。这些参数使您可以轻松地 AWS Ground Station 立即使用这些卫星。使用此模板 AWS Ground Station 时，您无需配置自己的值即可使用。但是，您可以自定义这些值以使模板适用于您的使用案例。

我可以在哪里接收我的数据？

数据流终端节点组设置为使用此模板的一部分创建的接收实例网络接口。接收器实例使用数据流端点应用程序从数据流端点定义 AWS Ground Station 的端口接收数据流。接收到数据后，可通过接收实

例的环回适配器上的 UDP 端口 50000 使用数据。有关设置数据流终端节点组的更多信息，请参阅。[AWS::GroundStation::DataflowEndpointGroup]

使用 AWS Ground Station 代理（宽带）的公共广播卫星

此示例建立在用户指南[JPSS-1-公共广播卫星 (PBS)-评估]部分所做的分析的基础上。

要完成此示例，您需要假设一个场景，即您要将 HRD 通信路径捕获为宽带数字中频 (digiF)，并在代理使用 SDR 在 Amazon EC2 实例 AWS Ground Station 上接收时对其进行处理。

> **Note**
>
> 实际的 JPSS HRD 通信路径信号的带宽为 30 MHz，但您需要将天线下行链路配置配置配置为将其视为 MHz带宽为 100 的信号，以便它可以流经正确的路径，供代理接收。 AWS Ground Station

沟通路径

本节介绍[规划您的数据流通信路径]入门。在本示例中，您需要在 AWS CloudFormation 模板中添加一个未在其他示例（Mappings 部分）中使用过的部分。

> **Note**
>
> 有关 AWS CloudFormation 模板内容的更多信息，请参阅[模板部分]。

首先，你要在 AWS CloudFormation 模板中为按区域划分 AWS Ground Station 的前缀列表设置一个“映射”部分。这样，Amazon EC2 实例安全组就可以轻松地引用前缀列表。有关使用前缀列表的更多信息，请参阅[使用 AWS Ground Station 代理配置 VPC]。

```
Mappings:
  PrefixListId:
    us-east-2:
      groundstation: pl-087f83ba4f34e3bea
    us-west-2:
      groundstation: pl-0cc36273da754ebdc
    us-east-1:
      groundstation: pl-0e5696d987d033653
```

```
    eu-central-1:
      groundstation: pl-03743f81267c0a85e
    sa-east-1:
      groundstation: pl-098248765e9effc20
    ap-northeast-2:
      groundstation: pl-059b3e0b02af70e4d
    ap-southeast-1:
      groundstation: pl-0d9b804fe014a6a99
    ap-southeast-2:
      groundstation: pl-08d24302b8c4d2b73
    me-south-1:
      groundstation: pl-02781422c4c792145
    eu-west-1:
      groundstation: pl-03fa6b266557b0d4f
    eu-north-1:
      groundstation: pl-033e44023025215c0
    af-south-1:
      groundstation: pl-0382d923a9d555425
```

在 “参数” 部分，您将添加以下参数。在通过 AWS CloudFormation 控制台创建堆栈时，您将为这些值指定值。

```
Parameters:
  EC2Key:
    Description: The SSH key used to access the EC2 receiver instance. Choose any
 SSH key if you are not creating an EC2 receiver instance. For instructions on how to
 create an SSH key see https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-
key-pairs.html
    Type: AWS::EC2::KeyPair::KeyName
    ConstraintDescription: must be the name of an existing EC2 KeyPair.

  AZ:
    Description: "The AvailabilityZone that the resources of this stack will be created
 in. (e.g. us-east-2a)"
    Type: AWS::EC2::AvailabilityZone::Name

  ReceiverAMI:
    Description: The Ground Station Agent AMI ID you want to use. Please note
 that AMIs are region specific. For instructions on how to retrieve an AMI
 see https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-
configuration.html#dataflows.ec2-configuration.amis
```

```
    Type: AWS::EC2::Image::Id
```

> **Note**
>
> 您需要创建密钥对，并提供 Amazon EC2 EC2Key 参数的名称。请参阅为您的 Amazon EC2 实例创建密钥对。
> 此外，在创建 AWS CloudFormation 堆栈时，您需要提供正确的特定于区域的 AMI ID。请参阅 AWS Ground Station Amazon 机器映像 (AMIs)。

其余的模板片段属于 AWS CloudFormation 模板的 “资源” 部分。

```
Resources:
  # Resources that you would like to create should be placed within the Resources
 section.
```

考虑到我们向 Amazon EC2 实例提供单一通信路径的场景，您知道您将拥有一条同步传输路径。根据本同步数据传输节，您必须使用 AWS Ground Station 代理设置和配置 Amazon EC2 实例，并创建一个或多个数据流终端节点组。首先，您需要为 AWS Ground Station 代理设置 Amazon VPC。

```
  ReceiverVPC:
    Type: AWS::EC2::VPC
    Properties:
      EnableDnsSupport: 'true'
      EnableDnsHostnames: 'true'
      CidrBlock: 10.0.0.0/16
      Tags:
      - Key: "Name"
        Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent VPC"
      - Key: "Description"
        Value: "VPC for EC2 instance receiving AWS Ground Station data"

  PublicSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref ReceiverVPC
      MapPublicIpOnLaunch: 'true'
      AvailabilityZone: !Ref AZ
      CidrBlock: 10.0.0.0/20
```

```
      Tags:
      - Key: "Name"
        Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent Public
Subnet"
      - Key: "Description"
        Value: "Subnet for EC2 instance receiving AWS Ground Station data"

  RouteTable:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: !Ref ReceiverVPC
      Tags:
        - Key: Name
          Value: AWS Ground Station Example - RouteTable

  RouteTableAssociation:
    Type: AWS::EC2::SubnetRouteTableAssociation
    Properties:
      RouteTableId: !Ref RouteTable
      SubnetId: !Ref PublicSubnet

  Route:
    Type: AWS::EC2::Route
    DependsOn: InternetGateway
    Properties:
      RouteTableId: !Ref RouteTable
      DestinationCidrBlock: '0.0.0.0/0'
      GatewayId: !Ref InternetGateway

  InternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: AWS Ground Station Example - Internet Gateway

  GatewayAttachment:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      VpcId: !Ref ReceiverVPC
      InternetGatewayId: !Ref InternetGateway
```

> **Note**
>
> 有关代理支持的 VPC 配置的更多信息，请参阅 AWS Ground Station AWS Ground Station 代理要求-VPC 图表。

接下来，您将设置 Receiver Amazon EC2 实例。

```
 # The placement group in which your EC2 instance is placed.
 ClusterPlacementGroup:
   Type: AWS::EC2::PlacementGroup
   Properties:
     Strategy: cluster

 # This is required for the EIP if the receiver EC2 instance is in a private subnet.
 # This ENI must exist in a public subnet, be attached to the receiver and be
associated with the EIP.
 ReceiverInstanceNetworkInterface:
   Type: AWS::EC2::NetworkInterface
   Properties:
     Description: Floating network interface
     GroupSet:
       - !Ref InstanceSecurityGroup
     SubnetId: !Ref PublicSubnet

 # An EIP providing a fixed IP address for AWS Ground Station to connect to. Attach it
to the receiver instance created in the stack.
 ReceiverInstanceElasticIp:
   Type: AWS::EC2::EIP
   Properties:
     Tags:
       - Key: Name
         Value: !Join [ "-" , [ "EIP" , !Ref "AWS::StackName" ] ]

 # Attach the ENI to the EC2 instance if using a separate public subnet.
 # Requires the receiver instance to be in a public subnet (SubnetId should be the id
of a public subnet)
 ReceiverNetworkInterfaceAttachment:
   Type: AWS::EC2::NetworkInterfaceAttachment
   Properties:
     DeleteOnTermination: false
     DeviceIndex: 1
```

```
      InstanceId: !Ref ReceiverInstance
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

  # Associate EIP with the ENI if using a separate public subnet for the ENI.
  ReceiverNetworkInterfaceElasticIpAssociation:
    Type: AWS::EC2::EIPAssociation
    Properties:
      AllocationId: !GetAtt [ReceiverInstanceElasticIp, AllocationId]
      NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

  # The EC2 instance that will send/receive data to/from your satellite using AWS
 Ground Station.
  ReceiverInstance:
    Type: AWS::EC2::Instance
    DependsOn: PublicSubnet
    Properties:
      DisableApiTermination: false
      IamInstanceProfile: !Ref GeneralInstanceProfile
      ImageId: !Ref ReceiverAMI
      AvailabilityZone: !Ref AZ
      InstanceType: c5.24xlarge
      KeyName: !Ref EC2Key
      Monitoring: true
      PlacementGroupName: !Ref ClusterPlacementGroup
      SecurityGroupIds:
        - Ref: InstanceSecurityGroup
      SubnetId: !Ref PublicSubnet
      Tags:
        - Key: Name
          Value: !Join [ "-" , [ "Receiver" , !Ref "AWS::StackName" ] ]
      # agentCpuCores list in the AGENT_CONFIG below defines the cores that the AWS
 Ground Station Agent is allowed to run on. This list can be changed to suit your use-
case, however if the agent isn't supplied with enough cores data loss may occur.
      UserData:
        Fn::Base64:
          Fn::Sub:
            - |
              #!/bin/bash
              yum -y update

              AGENT_CONFIG_PATH="/opt/aws/groundstation/etc/aws-gs-agent-config.json"
              cat << AGENT_CONFIG > "$AGENT_CONFIG_PATH"
              {
                "capabilities": [
```

```
                  "arn:aws:groundstation:${AWS::Region}:${AWS::AccountId}:dataflow-
endpoint-group/${DataflowEndpointGroupId}"
                ],
                "device": {
                  "privateIps": [
                    "127.0.0.1"
                  ],
                  "publicIps": [
                    "${EIP}"
                  ],
                  "agentCpuCores": [

 24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,72,73,74,75,76,77,78,79,80,81,8
                  ]
                }
              }
              AGENT_CONFIG

              systemctl start aws-groundstation-agent
              systemctl enable aws-groundstation-agent

              # <Tuning Section Start>
              # Visit the AWS Ground Station Agent Documentation in the User Guide for
 more details and guidance updates

              # Set IRQ affinity with list of CPU cores and Receive Side Scaling mask
              # Core list should be the first two cores (and hyperthreads) on each
 socket
              # Mask set to everything currently
              # https://github.com/torvalds/linux/blob/v4.11/Documentation/networking/
scaling.txt#L80-L96
              echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh '0 1 48
 49' 'ffffffff,ffffffff,ffffffff' >>/var/log/user-data.log 2>&1" >>/var/spool/cron/root

              # Reserving the port range defined in the GS agent ingress address in
 the Dataflow Endpoint Group so the kernel doesn't steal any of them from the GS agent.
 These ports are the ports that the GS agent will ingress data
              # across, so if the kernel steals one it could cause problems ingressing
 data onto the instance.
              echo net.ipv4.ip_local_reserved_ports="42000-50000" >> /etc/sysctl.conf

              # </Tuning Section End>

              # We have to reboot for linux kernel settings to apply
```

```
              shutdown -r now

            - DataflowEndpointGroupId: !Ref DataflowEndpointGroup
              EIP: !Ref ReceiverInstanceElasticIp
```

```
 # The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
 # Station will use to send/receive data to/from your satellite.
 DataflowEndpointGroup:
   Type: AWS::GroundStation::DataflowEndpointGroup
   Properties:
     ContactPostPassDurationSeconds: 180
     ContactPrePassDurationSeconds: 120
     EndpointDetails:
       - AwsGroundStationAgentEndpoint:
           Name: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
           EgressAddress:
             SocketAddress:
               Name: 127.0.0.1
               Port: 55000
           IngressAddress:
             SocketAddress:
               Name: !Ref ReceiverInstanceElasticIp
               PortRange:
                 Minimum: 42000
                 Maximum: 55000
```

您还需要相应的策略、角色和配置文件，以便 AWS Ground Station 在您的账户中创建 elastic network interface (ENI)。

```
 # The security group for your EC2 instance.
 InstanceSecurityGroup:
   Type: AWS::EC2::SecurityGroup
   Properties:
     GroupDescription: AWS Ground Station receiver instance security group.
     VpcId: !Ref ReceiverVPC
     SecurityGroupEgress:
       - CidrIp: 0.0.0.0/0
         Description: Allow all outbound traffic by default
```

```
          IpProtocol: "-1"
      SecurityGroupIngress:
        # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
        - IpProtocol: udp
          Description: Allow AWS Ground Station Incoming Dataflows
          ToPort: 50000
          FromPort: 42000
          SourcePrefixListId:
            Fn::FindInMap:
              - PrefixListId
              - Ref: AWS::Region
              - groundstation

   # The EC2 instance assumes this role.
  InstanceRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Principal:
              Service:
                - "ec2.amazonaws.com"
            Action:
              - "sts:AssumeRole"
      Path: "/"
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
        - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
        - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
        - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM
        - arn:aws:iam::aws:policy/AWSGroundStationAgentInstancePolicy
      Policies:
        - PolicyDocument:
            Statement:
              - Action:
                  - sts:AssumeRole
                Effect: Allow
                Resource: !GetAtt GroundStationKmsKeyRole.Arn
            Version: "2012-10-17"
          PolicyName: InstanceGroundStationApiAccessPolicy
```

```
  # The instance profile for your EC2 instance.
  GeneralInstanceProfile:
    Type: AWS::IAM::InstanceProfile
    Properties:
      Roles:
        - !Ref InstanceRole

  # The IAM role that AWS Ground Station will assume to access and use the KMS Key for
 data delivery
  GroundStationKmsKeyRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Statement:
          - Action: sts:AssumeRole
            Effect: Allow
            Principal:
              Service:
                - groundstation.amazonaws.com
            Condition:
              StringEquals:
                "aws:SourceAccount": !Ref AWS::AccountId
              ArnLike:
                "aws:SourceArn": !Sub "arn:${AWS::Partition}:groundstation:
${AWS::Region}:${AWS::AccountId}:mission-profile/*"
          - Action: sts:AssumeRole
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"

  GroundStationKmsKeyAccessPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Statement:
          - Action:
              - kms:Decrypt
            Effect: Allow
            Resource: !GetAtt GroundStationDataDeliveryKmsKey.Arn
      PolicyName: GroundStationKmsKeyAccessPolicy
      Roles:
        - Ref: GroundStationKmsKeyRole

  GroundStationDataDeliveryKmsKey:
```

```
    Type: AWS::KMS::Key
    Properties:
      KeyPolicy:
        Statement:
          - Action:
              - kms:CreateAlias
              - kms:Describe*
              - kms:Enable*
              - kms:List*
              - kms:Put*
              - kms:Update*
              - kms:Revoke*
              - kms:Disable*
              - kms:Get*
              - kms:Delete*
              - kms:ScheduleKeyDeletion
              - kms:CancelKeyDeletion
              - kms:GenerateDataKey
              - kms:TagResource
              - kms:UntagResource
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"
            Resource: "*"
          - Action:
              - kms:Decrypt
              - kms:GenerateDataKeyWithoutPlaintext
            Effect: Allow
            Principal:
              AWS: !GetAtt GroundStationKmsKeyRole.Arn
            Resource: "*"
            Condition:
              StringEquals:
                "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
              ArnLike:
                "kms:EncryptionContext:sourceArn": !Sub "arn:
${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
          - Action:
              - kms:CreateGrant
            Effect: Allow
            Principal:
              AWS: !Sub "arn:${AWS::Partition}:iam::${AWS::AccountId}:root"
            Resource: "*"
            Condition:
```

```
              ForAllValues:StringEquals:
                "kms:GrantOperations":
                  - Decrypt
                  - GenerateDataKeyWithoutPlaintext
                "kms:EncryptionContextKeys":
                  - sourceArn
                  - sourceAccount
              ArnLike:
                "kms:EncryptionContext:sourceArn": !Sub "arn:
${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
              StringEquals:
                "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
        Version: "2012-10-17"
      EnableKeyRotation: true
```

AWS Ground Station 配置

本节介绍创建配置入门。

你需要一个跟踪配置来设置你使用自动追踪的偏好。选择 P REFERRED 作为自动跟踪可以提高信号质量，但由于 JPSS-1 星历质量足够，因此不需要满足信号质量。

```
  TrackingConfig:
    Type: AWS::GroundStation::Config
    Properties:
      Name: "JPSS Tracking Config"
      ConfigData:
        TrackingConfig:
          Autotrack: "PREFERRED"
```

根据通信路径，你需要定义一个天线下行链路配置来表示卫星部分，还需要定义一个数据流端点配置来引用定义端点详细信息的数据流端点组。

```
  # The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
 used to
  # downlink data from your satellite.
  SnppJpssDownlinkDigIfAntennaConfig:
    Type: AWS::GroundStation::Config
```

```
    Properties:
      Name: "SNPP JPSS Downlink WBDigIF Antenna Config"
      ConfigData:
        AntennaDownlinkConfig:
          SpectrumConfig:
            Bandwidth:
              Units: "MHz"
              Value: 100
            CenterFrequency:
              Units: "MHz"
              Value: 7812
            Polarization: "RIGHT_HAND"

  # The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
  # from your satellite.
  DownlinkDigIfEndpointConfig:
    Type: AWS::GroundStation::Config
    Properties:
      Name: "Aqua SNPP JPSS Terra Downlink DigIF Endpoint Config"
      ConfigData:
        DataflowEndpointConfig:
          DataflowEndpointName: !Join [ "-" , [ !Ref "AWS::StackName" , "Downlink" ] ]
          DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station 任务简介

本节介绍创建任务档案入门。

现在你已经有了相关的配置，你可以用它们来构造数据流。其余参数将使用默认值。

```
  # The AWS Ground Station Mission Profile that groups the above configurations to
define how to
  # uplink and downlink data to your satellite.
  SnppJpssMissionProfile:
    Type: AWS::GroundStation::MissionProfile
    Properties:
      Name: !Sub 'JPSS WBDigIF gs-agent EC2 Delivery'
      ContactPrePassDurationSeconds: 120
      ContactPostPassDurationSeconds: 120
      MinimumViableContactDurationSeconds: 180
```

```
      TrackingConfigArn: !Ref TrackingConfig
      DataflowEdges:
        - Source: !Ref SnppJpssDownlinkDigIfAntennaConfig
          Destination: !Ref DownlinkDigIfEndpointConfig
      StreamsKmsKey:
        KmsKeyArn: !GetAtt GroundStationDataDeliveryKmsKey.Arn
      StreamsKmsRole: !GetAtt GroundStationKmsKeyRole.Arn
```

把它放在一起

利用上述资源，您现在可以安排 JPSS-1 联系人，以便从任何已上线人员同步传送数据。 AWS Ground Station AWS Ground Station 地点

以下是一个完整的 AWS CloudFormation 模板，其中包括本节中描述的所有资源，这些资源组合成一个可以直接在中使用的模板 AWS CloudFormation。

名`DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml`为的 AWS CloudFormation 模板旨在让你快速访问开始接收 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 卫星的数字化中频 (digiF) 数据。它包含一个 Amazon EC2 实例和使用 AWS Ground Station 代理接收原始的 digiF 直接广播数据所需的 AWS CloudFormation 资源。

如果你的账户未登录 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra，请参阅。机载卫星

> **Note**
>
> 您可以使用有效 AWS 凭证访问客户登录 Amazon S3 存储桶，从而访问该模板。以下链接使用区域性 Amazon S3 存储桶。更改`us-west-2`区域代码以表示要在其中创建 AWS CloudFormation 堆栈的相应区域。
> 此外，以下指令使用 YAML。但是，模板有 YAML 和 JSON 这两种格式。要使用 JSON，请在下载模板`.json`时将`.yml`文件扩展名替换为。

要使用下载模板 AWS CLI，请使用以下命令：

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

在浏览器中导航到以下 URL，可以在控制台中查看和下载此模板：

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

您可以使用以下链接直接在中 AWS CloudFormation 指定模板：

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

该模板还定义了哪些其他资源？

该DirectBroadcastSatelliteWbDigIfEc2DataDelivery模板包括以下其他资源：

- 接收方实例弹性网络接口-（视情况而定）在指定的子网中创建弹性网络接口（PublicSubnetId如果提供）。如果接收方实例位于私有子网中，则需要这样做。elastic network interface 将与 EIP 关联并连接到接收方实例。
- 接收器实例弹性 IP- AWS Ground Station 将连接到的弹性 IP。这会连接到接收器实例或 elastic network 接口。
- 以下弹性 IP 关联之一：
  - 接收器实例与弹性 IP 关联-弹性 IP 与您的接收器实例的关联（如果PublicSubnetId未指定）。这需要SubnetId引用公有子网。
  - 接收方实例与弹性 IP 关联的弹性网络接口-弹性 IP 与接收方实例弹性网络接口的关联（PublicSubnetId如果已指定）。
- （可选）CloudWatch 事件触发器-使用联系 AWS Ground Station 前后发送 CloudWatch 的事件触发的 AWS Lambda 函数。该 AWS Lambda 函数将启动并有选择地停止您的接收器实例。
- （可选）亚马逊联系人 EC2 验证-使用 Lambda 为带有 SNS 通知的联系人设置亚马逊 EC2 实例的验证系统的选项。需要注意的是，这可能会产生费用，具体取决于您当前的使用情况。
- 其他任务概况 ——其他公共广播卫星（Aqua、SNPP 和 Terra）的任务概况。
- 其他天线下行链路配置——其他公共广播卫星（Aqua、SN PP 和 Terra）的天线下行链路配置。

已填充此模板中卫星的值和参数。这些参数使您可以轻松地 AWS Ground Station 立即使用这些卫星。使用此模板 AWS Ground Station 时，您无需配置自己的值即可使用。但是，您可以自定义这些值以使模板适用于您的使用案例。

我可以在哪里接收我的数据？

数据流终端节点组设置为使用此模板的一部分创建的接收实例网络接口。接收器实例使用 AWS Ground Station 代理从数据流端 AWS Ground Station 点定义的端口接收数据流。有关设置数据流终

端节点组的更多信息，请参阅。 AWS::GroundStation::DataflowEndpointGroup有关 AWS Ground Station 代理的更多信息，请参阅什么是代 AWS Ground Station 理？

故障排除

以下文档可以帮助您解决使用时可能出现的问题 AWS Ground Station。

主题

对向 Amazon 传送数据的联系人进行故障排除 EC2

如果您无法成功完成 AWS Ground Station 联系，则需要验证您的Amazon EC2 实例是否正在运行，验证您的数据流终端节点应用程序是否正在运行，并验证您的数据流终端节点应用程序的流配置是否正确。

> **Note**
>
> DataDefender (DDX) 是目前支持的数据流端点应用程序的示例 AWS Ground Station

先决条件

以下过程假设已经设置了 Amazon EC2 实例。要在中设置 Amazon EC2 实例 AWS Ground Station，请参阅入门。

步骤 1：验证您的 EC2 实例是否正在运行

以下过程说明如何在控制台中找到您的 Amazon EC2 实例，并在该实例未运行时将其启动。

1. 找到用于您正在进行故障诊断的联系的 Amazon EC2 实例。使用以下步骤：

    a. 在AWS CloudFormation控制面板中，选择包含您的 Amazon EC2 实例的堆栈。

b. 选择 “资源” 选项卡，然后在 “逻辑 ID” 列中找到您的 Amazon EC2 实例。在状态列中验证实例是否已创建。

c. 在 “物理 ID” 列中，选择您的 Amazon EC2 实例的链接。这将带您进入Amazon EC2 管理控制台。

2. 在亚马逊 EC2 管理控制台中，确保您的亚马逊 EC2 实例状态处于运行状态。

3. 如果您的实例正在运行，请继续执行下一步。如果您的实例没有运行，请使用以下步骤启动实例：

   - 选择您的 Amazon EC2 实例后，选择操作 > 实例状态 > 启动。

步骤 2：确定使用的数据流应用程序的类型

如果您使用AWS Ground Station 代理进行数据传输，请重定向至 “故障排除 AWS Ground Station 代理” 部分。否则，如果您使用的是 DataDefender (DDX) 应用程序，请继续。the section called “步骤 3：验证数据流应用程序是否正在运行”

步骤 3：验证数据流应用程序是否正在运行

验证状态 DataDefender 需要您连接到 Amazon 中的实例 EC2。有关连接到您的实例的更多详细信息，请参阅 Connect 到您的 Linux 实例。

以下过程提供了在 SSH 客户端中使用命令进行故障排除的步骤。

1. 打开终端或命令提示符并使用 SSH 连接到您的 Amazon EC2 实例。转发远程主机的端口 80 以查看 DataDefender Web 用户界面。以下命令演示如何使用 SSH 通过启用端口转发的堡垒连接到 Amazon EC2 实例。

   > **Note**
   >
   > 您必须<SSH KEY><BASTION HOST><HOST>使用您的特定 SSH 密钥、堡垒主机名和 Amazon EC2 实例主机名替换、和。

   对于 Windows：

   ```
   ssh -L 8080:localhost:80 -o ProxyCommand="C:\Windows\System32\OpenSSH\ssh.exe -o
    \"ForwardAgent yes\" -W %h:%p -i \"<SSH KEY>\" ec2-user@<BASTION HOST>" -i "<SSH
    KEY>" ec2-user@<HOST>
   ```

适用于 Mac

```
ssh -L 8080:localhost:80 -o ProxyCommand="ssh -A -o 'ForwardAgent yes' -W %h:%p -i
 <SSH KEY> ec2-user@<BASTION HOST>" -i <SSH  KEY> ec2-user@<HOST>
```

2. 通过对输出 DataDefender 中名为 ddx 的正在运行的进程执行 greping（检查）来验证（也称为 DDX）是否正在运行。下面提供了针对正在运行的进程和成功的示例输出的 grep（检查）命令。

```
[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
```

如果 DataDefender 正在运行，请跳至 “the section called “步骤 4：验证您的数据流应用程序流是否已配置”否则”，继续下一步。

3. 开始 DataDefender 使用如下所示的命令。

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

如果在使用命令后 DataDefender 正在运行，请跳至 “the section called “步骤 4：验证您的数据流应用程序流是否已配置”否则”，继续下一步。

4. 使用以下命令检查以下文件，以查看安装和配置时是否存在任何错误 DataDefender。

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

> **Note**
>
> 检查这些文件时发现的一个常见问题是，您的亚马逊 EC2 实例所运行的 Amazon VPC 无法访问 Amazon S3 来下载安装文件。如果您在日志中发现这是问题所在，请检查您的 EC2 实例的 Amazon VPC 和安全组设置，确保它们不会阻止对 Amazon S3 的访问。

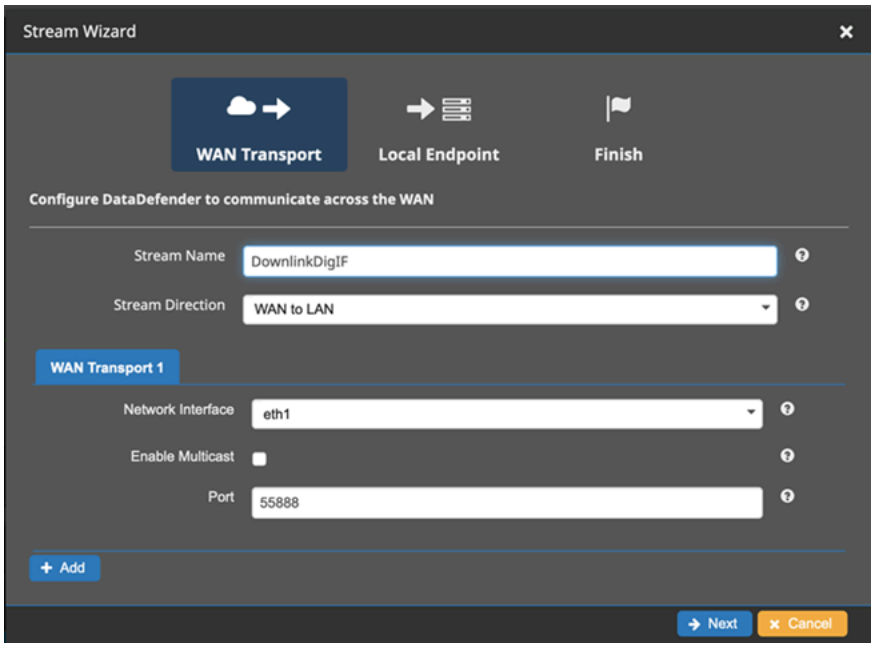
如果在检查您 DataDefender 的 Amazon VPC 设置后正在运行，请继续the section called “步骤 4：验证您的数据流应用程序流是否已配置”。如果问题仍然存在，请联系 AWS Support，并发送包含问题描述的日志文件。

步骤 4：验证您的数据流应用程序流是否已配置

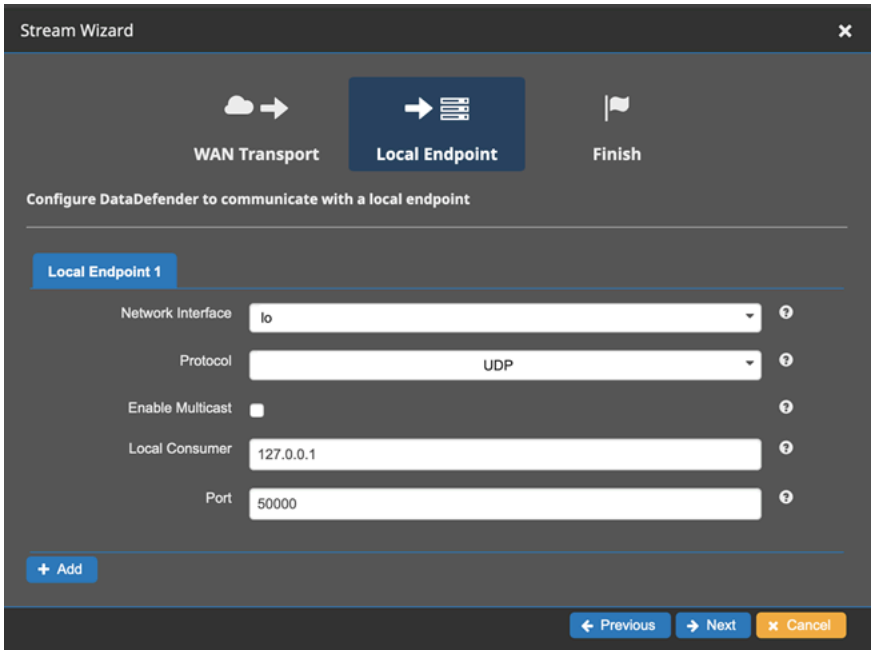
1. 在网络浏览器中，通过在地址栏中输入以下地址来访问您的 DataDefender 网络用户界面：localhost: 8080。然后按 Enter。
2. 在DataDefender仪表板上，选择 “转到详细信息”。
3. 从流列表中选择您的流，然后选择编辑流。
4. 在流向导对话框中，执行以下操作：

   a. 在广域网传输窗格中，确保将流方向选为广域网到局域网。

   b. 在端口框中，确保您为数据流端点组选择的 WAN 端口已存在。默认情况下，此端口为 55888。然后选择下一步。



   c. 在本地终端节点窗格中，确保端口框中存在有效端口。默认情况下，此端口为 50000。从 AWS Ground Station 服务收到数据后 DataDefender ，您将在此端口上接收数据。然后选择下一步。



d. 如果您更改了任何值，请通过其余菜单选择下一步。另外，您也可以取消流向导菜单。

现在，您已确保您的 Amazon EC2 实例和 DataDefender 均处于运行状态，并且配置正确，可以从中接收数据 AWS Ground Station。继续the section called “步骤 5：确保您的接收器实例子网中有足够的可用的 IP 地址”。

步骤 5：确保您的接收器实例子网中有足够的可用的 IP 地址

以下过程说明如何在控制台中查找 Amazon EC2 接收器实例中可用 IP 地址的数量。

1. 对于用于联系的每个 Amazon EC2 接收器实例，您正在进行故障排除。使用以下步骤：

   a. 在AWS CloudFormation控制面板中，选择包含您的 Amazon EC2 实例的堆栈。

   b. 选择 “资源” 选项卡，然后在 “逻辑 ID” 列中找到您的 Amazon EC2 实例。在状态列中验证实例是否已创建。

   c. 在 “物理 ID” 列中，选择您的 Amazon EC2 实例的链接。这将带您进入Amazon EC2 管理控制台。

2. 在亚马逊 EC2 管理控制台中，找到并单击您的亚马逊 EC2 接收器实例的实例摘要中的子网 ID 链接。这将带您进入相应的 Amazon VPC 管理控制台。

3. 在 Amazon VPC 管理控制台中选择匹配的子网，然后查看子网的详细信息以了解可用 IPv4 地址。如果此数字不及使用此 Amazon EC2 接收器实例的 dataflow 终端节点的数量，请执行以下操作：

a. 更新 AWS CloudFormation 模板的相应子网CidrBlock以使其大小正确。有关子网大小的更多详细信息，请参阅子网 CIDR 块。

b. 使用更新的 AWS CloudFormation 模板重新部署堆栈。

如果您仍然遇到问题，请联系 AWS Support。

解决失败的联系人问题

当 AWS Ground Station 检测到您的资源配置存在问题时，联系人的终端联系状态将为 FAILED。下面提供了可能导致联络失败的常见用例，以及帮助排除故障的步骤。

> **Note**
>
> 本指南专门针对失败的联系状态，不适用于其他失败状态，例如、或 FAI LED_TO_SCH AWS_FAILEDED AWS_CANCELLEDULE。有关联络状态的更多信息，请参阅the section called “AWS Ground Station 联系人状态”

数据流端点失败的用例

以下是可能导致基于数据流端点的数据流的联系失败状态的常见用例列表：

- Dataflow 端点从未连接 —— AWS Ground Station 天线和你的 Dataflow 端点组之间一个或多个数据流的连接从未建立。
- Dataflow 端点连接较晚- AWS Ground Station 天线和您的 Dataflow 端点组之间用于一个或多个数据流的连接是在联系开始时间之后建立的。
- Dataflow 终端节点的子网没有可用的 IP 地址- AWS Ground Station由于接收器实例的子网中没有任何可用的 IP 地址，因此的数据传输解决方案无法在您的私有网络中创建 ENI。

对于任何数据流端点故障案例，建议调查以下内容：

- 在联系开始时间之前，确认接收者 Amazon EC2 实例已成功启动。
- 确认联系期间数据流端点软件已启动并正在运行。
- 确保每个接收方实例子网的每个数据流端点至少有一个可用的 IP 地址。

有关更具体的故障排除步骤，请参阅有关对向 Amazon 传送数据的联系人进行故障排除 EC2的部分。

AWS Ground Station 代理失败的用例

以下是可能导致基于代理的数据流出现失败联络状态的常见用例列表：

- AWS Ground Station 代理从未报告过状态-负责在您的 Dataflow Endpoint Group 上协调一个或多个数据流的数据传输的代理从未成功向其报告状态。 AWS Ground Station此状态更新应在联络结束后的几秒钟内发生。
- AWS Ground Station 代理启动较晚-负责在您的 Dataflow Endpoint Group 上为一个或多个数据流协调数据交付的代理启动得很晚，也就是联系开始时间之后。

对于任何 A AWS Ground Station gent 数据流失败案例，建议调查以下内容：

- 在联系开始时间之前，确认接收者 Amazon EC2 实例已成功启动。
- 确认在启动时和联络期间代理应用程序已启动并正在运行。
- 在联系结束后 15 秒内确认代理应用程序和 Amazon EC2 实例未关闭。这为代理提供了足够的时间向 AWS Ground Station报告状态。

有关更具体的故障排除步骤，请参阅有关对向 Amazon 传送数据的联系人进行故障排除 EC2的部分。

排查计划失败的联系人故障

当 AWS Ground Station 检测到您的资源配置或内部系统存在问题时，联系人将以 FAILED_TO_SCHEDULE 状态结束。以 FAILE D_TO_SCHEDULE 状态结束的联系人可以选择提供其他背景信息。`errorMessage`有关描述联系人的信息，请参阅 DescribeContactAPI。

下面提供了可能导致 FAILED_TO_SCHEDULE 联系的常见用例，以及帮助进行故障排除的步骤。

> **Note**
>
> 本指南专门针对 FAILED_TO_SCHED ULE 联系状态，不适用于其他故障状态，例如、或 FAILED。AWS_FAILEDAWS_CANCELLED有关联络状态的更多信息，请参阅the section called “AWS Ground Station 联系人状态”

不支持 Antenna Downlink Demod Decode Config 中指定的设置

用于安排此次联系的任务配置文件antenna-downlink-demod-decode 中的配置无效。

先前存在的 AntennaDownlinkDemodDecode 配置

- 如果您的 antenna-downlink-demod-decode配置最近发生了更改，请在尝试安排之前回滚到以前运行的版本。
- 如果这是对现有配置的故意更改，或者是以前存在的配置无法成功调度，请按照下一步操作如何载入新 AntennaDownlinkDemodDecode 配置。

新创建的 AntennaDownlinkDemodDecode 配置

请 AWS Ground Station 直接联系以加入您的新配置。使用 AWS Supp ort 创建案例`contactId`，包括以 FAILED_TO_SCHED ULE 状态结束的案例

一般故障排除步骤

如果上述故障排除步骤未能解决您的问题：

- 使用相同的任务配置文件重新尝试安排联系人或安排其他联系人。有关如何预约联系人的信息，请参阅ReserveContact。
- 如果您继续收到此任务档案的 FAILED_TO_SCHEDULE 状态，请联系 AWS Support

排除 DataflowEndpointGroups 未处于正常状态的故障

下面列出了您的数据流端点组可能未处于 `HEALTHY` 状态的原因以及需要采取的适当纠正措施。

- `NO_REGISTERED_AGENT`-启动您的 EC2 实例，该实例将注册代理。请注意，您必须拥有有效的控制器配置文件才能成功实现本次调用。有关配置该文件的详细信息，请参阅 使用 AWS Ground Station 代理。
- `INVALID_IP_OWNERSHIP`-使用 DeleteDataflowEndpointGroup API 删除 Dataflow 端点组，然后使用 CreateDataflowEndpointGroup API 使用与实例关联的 IP 地址和端口重新创建 Dataflow 端点组。EC2
- `UNVERIFIED_IP_OWNERSHIP`：IP 地址尚未经过验证。验证会定期进行，因此这个问题应该会自行解决。

- NOT_AUTHORIZED_TO_CREATE_SLR：账户没有创建必要的服务相关角色的权限。请查看 在 Ground Station 中使用与服务相关的角色 中的故障排除步骤

对无效的星历进行故障排除

将自定义星历上传到 AWS Ground Station 它时，要经过异步验证工作流程，然后才变成。ENABLED此流程可确保卫星标识符、元数据和轨迹有效。

当星历验证失败时，DescribeEphemeris将返回 EphemerisInvalidReason，这可以深入了解星历验证失败的原因。的潜在值EphemerisInvalidReason如下：

值	描述	故障排除操作
METADATA_INVALID	提供的航天器标识符（例如卫星 ID）无效	检查星历数据中提供的 NORAD ID 或其他标识符
TIME_RANGE_INVALID	所提供星历的开始、结束或到期时间无效	确保“开始时间”早于“现在”（建议将开始时间设置为几分钟前），“结束时间”晚于“开始时间”，并且“结束时间”在“到期时间”之后
TRAJECTORY_INVALID	所提供的星历定义无效的航天器轨迹	确认提供的轨迹是连续的，并且针对正确的卫星。
VALIDATION_ERROR	处理用于验证的星历时出现内部服务错误	重试上传

以下提供对 INVALID 星历的示例响应 DescribeEphemeris：

```
{
  "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"
      }
    }
  },
}
```

> **Note**
>
> 如果星历的状态为ERROR，则星历不是ENABLED由于服务问题造成的。 AWS Ground Station 您应该尝试通过再次提供星历。CreateEphemerisENABLED如果问题是暂时的，则新的星历可能会变成。

> **Note**
>
> AWS Ground Station 将星历视为个性化使用数据。如果您使用此可选功能，AWS 将使用您的星历数据来提供故障排除支持。

对未收到任何数据的联系人进行故障排除

联系人可能显示为成功，但仍未收到任何数据。这可能意味着您收到的是空的 PCAP 文件，或者如果您使用 S3 数据传输，则根本没有 PCAP 文件。这可能是由于多种原因造成。以下内容讨论了其中一些原因以及如何解决这些问题。

下行链路配置不正确

从卫星接收数据的每个联系人都将有一个关联的天线下行传输配置或天线下行传输解调解码配置 。如果指定的配置与卫星传输的信号不一致，则 AWS Ground Station 将无法接收传输的信号。这将导致未收到任何数据 AWS Ground Station。

要解决此问题，请验证您使用的配置是否与卫星传输的信号一致。例如，验证您是否设置了正确的中心频率、带宽、极化以及解调和解码参数（如果需要）。

卫星操纵

有时，卫星可能会执行暂时禁用其某些通信系统的操纵。该演习还可能显著改变卫星在天空中的位置。AWS Ground Station 将无法接收来自未传输信号的卫星的信号，或者如果使用的星历使 AWS Ground Station 天线指向天空中不存在卫星的位置。

如果您正在尝试与 NOAA 运营的公共广播卫星进行通信，则可以在 NOAA 卫星警报消息页面上找到描述中断或操纵的消息。该消息可能包括预计何时恢复数据传输的时间表，也可能在随后的消息中发布。

如果你正在与自己的卫星通信，你有责任了解自己的卫星运行情况，以及这会如何影响与之通信 AWS Ground Station。如果您正在执行会影响卫星轨迹的操作，则可能包括提供更新的自定义星历数据。有关提供自定义星历数据的更多信息，请参阅。提供自定义星历数据

AWS Ground Station 中断

如果 AWS Ground Station 导致联系失败或取消，则 AWS Ground Station 会将联系状态设置为AWS_FAILED、或AWS_CANCELLED。有关联系人生命周期的更多信息，请参阅了解联系人生命周期。在某些情况下， AWS Ground Station 可能会出现故障，导致数据无法发送到您的账户，但不会导致联系人处于 “AWS_FAILED或 AWS_CANCELLED” 状态。发生这种情况时， AWS Ground Station 应在您的 Healt AWS h 控制面板上发布特定于账户的事件。有关 Healt AWS h 控制面板的更多信息，请参阅 AWS Health 用户指南。

限额和限制

您可以查看支持的区域、其关联的终端节点以及终端节点和配额AWS Ground Station 的配额。

如果需要，您可以使用 Service Quotas 控制台、AWS API 和 AWS CLI 请求提高配额。

服务条款

有关 AWS Ground Station 服务条款，请参阅 AWS 服务条款。

《 AWS Ground Station 用户指南》的文档历史记录

下表描述了每个版本的《 AWS Ground Station 用户指南》中的重要更改。

变更	说明	日期
文档更新	添加了对已配置资源的联系人利用率的说明。	2025 年 4 月 4 日
新特征	更新了用户指南，加入了 AWS Ground Station 数字双胞胎。	2024 年 8 月 6 日
文档更新	更新了用户指南的许多部分，包括新的图表、示例等。	2024 年 7 月 18 日
文档更新	在《用户指南》中添加了 RSS 提要。	2024 年 7 月 18 日
文档更新	将 AWS Ground Station 代理用户指南拆分为单独的用户指南。	2024 年 7 月 18 日
新特征	现在，可以在可见时间范围之外安排最多 30 秒的联系人。可见性时间包含在 DescribeContact 响应中。	2024 年 3 月 26 日
文档更新	改进了组织结构，并添加了“EC2 实例选择和 CPU 规划”部分。	2024 年 3 月 6 日
文档更新	在《 AWS Ground Station 代理用户指南》中添加了与代理一起运行服务和进程的新最佳实践。 AWS Ground Station	2024 年 2 月 23 日
文档更新	添加了代理版本说明页面。	2024 年 2 月 21 日

模板更新	在 DirectBroadcastSatelliteWbDigIfEc 2 DataDelivery 模板中添加了对单独公有子网的支持。	2024 年 2 月 14 日
文档更新	用户通知服务 在监控文档中添加了对 AWS 的引用。	2023年8月6日
文档更新	添加了使用要在 AWS Ground Station 控制台中显示的名称来标记卫星的说明。	2023 年 7 月 26 日
新特征	为发布宽带 digiF 数据传输添加了《 AWS Ground Station 代理用户指南》	2023 年 4 月 12 日
新的 AWS 托管策略	AWS Ground Station 添加了一个名为的新策略 AWSGroundStationAgentInstancePolicy。	2023 年 4 月 12 日
新特征	更新了 CPE 预览版发布的用户指南。	2022 年 11 月 9 日
新的 AWS 托管策略	AWS Ground Station 添加了 AWSService RoleForGroundStationDataflowEndpointGroup service-linked-role (SLR)，其中包含一个名为 AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy的新策略。	2022 年 11 月 2 日
新特征	更新了用户指南，增加了与的集成 AWS CLI。	2020 年 4 月 17 日
新特征	更新了用户指南，增加了与 CloudWatch 指标的集成。	2020 年 2 月 24 日

新建模板	《AWS Ground Station 用户指南》中添加了公共广播卫星（AquaSnppJpss 模板）。	2020 年 2 月 19 日
新特征	更新了用户指南以包含跨区域数据传输。	2020 年 2 月 5 日
文档更新	更新了 AWS Ground Station 使用 CloudWatch 事件进行监控的示例和描述。	2020 年 2 月 4 日
文档更新	模板位置已更新且“入门”和 “故障排除”部分已修订。	2019 年 12 月 19 日
新的故障排除部分	故障排除部分添加至 AWS Ground Station 用户指南。	2019 年 11 月 7 日
新入门主题	更新了入门主题，其中包括最新的 AWS CloudFormation 模板。	2019 年 7 月 1 日
Kindle 版本	发布了 AWS Ground Station 用户指南 Kindle 版本。	2019 年 6 月 20 日
新服务和指南	这是《AWS Ground Station 用户指南》 AWS Ground Station 的初始版本。	2019 年 5 月 23 日

AWS 词汇表

有关最新 AWS 术语，请参阅《AWS 词汇表 参考资料》中的AWS 词汇表。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。